Position Determination of Mobile Unit Based on Inertial Navigation System

YIP, Wai Lee

A Thesis Submitted in Partial Fulfillment
of the Requirements for the
Degree of Master of Philosophy
In
Automation and Computer-Aided Engineering

September 2008

Thesis/Assessment Committee

Professor Liu Yun-Hui (Chair)

Professor Yam Yeung (Thesis Supervisor)

Professor Li Wen Jung (Committee Member)

Professor Huang Jen-Kuang (External Examiner)

Abstract

Nowadays, positioning technologies are implemented and utilized in many diverse applications. The most common and well-known positioning system is Global Positioning System (GPS). This globally available system is well suited for many outdoor positioning tasks. However, GPS has the signal blockage problem in dense areas and indoor. On the other hand, Inertial Navigation System (INS) is a self-contained device which does not require external signal. Therefore it can provide complementary support to GPS. The objective of this thesis is to build a system to determine the movement of mobile unit based on INS.

The work is started by investigating and evaluating the performance of the Micro Electro-Mechanical Systems (MEMS) accelerometer in position determination. A one dimensional INS is built to test the accelerometer. To enhance the performance of the calculated displacement, a fuzzy logic based steady state detector is proposed to reduce the noise when the unit is stationary. Furthermore, a Kalman filter is implemented to reduce the random noise when the unit is moving. The performance of the noise reduction methods are verified by experimental results.

Then, a Gyroscope-Free-INS (GF-INS) is constructed with six MEMS accelerometers based on a particular cube-shape configuration. To improve the performance, modified configuration error compensation is implemented, based upon which the configuration error of the GF-INS is identified and compensated. Another noise reduction method is fuzzy rule based motion state detector. It aims to determine the state of the motion of the mobile unit so that constraints can be added in the integration process. Gradient descent is proposed to train the fuzzy membership functions to enhance the accuracy of the detector. Simulations and experiment results are included to illustrate the performance of the error reduction methods.

摘要

現今，定位技術被實施及應用在很多不同的用途上。其中最常見及眾所周知的定位系統就是全球定位系統 (GPS)。這個全球可用的系統對於很多戶外定位任務是非常合適的。然而，GPS 在樓宇稠密地區及室內存在著訊號被阻礙的問題。另一方面，慣性導航系統 (INS) 是一個不需外部訊號的獨立性設備，所以它能給 GPS 提供補給支援。本論文的宗旨是根據 INS 的原理建立一個系統用以測量流動物件的移動。

工作開始於研究及評估微機電系統加速度傳感器在測量移動位置上的表現。一個一維度的 INS 被建立並用以測試該加速度傳感器。爲了增加計算位移結果的準確性，一個基於模糊邏輯的不動探測器被提議，用以減少流動物件不動時的噪音。此外，一個卡爾曼濾波器被建立，用以減少物件移動時的隨機噪音。這些噪音降低方法的表現由實驗結果核實。

然後，一個由六個微機電加速度傳感器組成及根據一特殊立方體配置的無陀螺儀慣性導航系統 (GF-INS) 被建立。爲了改進其表現，經修改的配置誤差補償方法被實施，GF-INS 的配置誤差被確認及補償。另一個噪音降低方法是一個基於模糊規則的移動狀態探測器，它的目的是測量移動物件的移動狀態，從而在積分計算中增加限制。梯度下降法被提議用於訓練模糊歸屬函數以增加探測器的準確性。模擬運算及實驗結果都包括在內用以說明這些噪音降低方法的表現。

Acknowledgement

The works presented in this thesis could not be accomplished without the support of my supervisor Prof. Yam Yeung. He gave me a great chance to enter the world of research. His patient guidance and encouragement were important element for me to complete the works. I would like to express my sincere thanks to him.

I would also like to thank Prof. Li Wen Jung, Prof. Liu Yun-Hui and Prof. Huang Jen-Kuang for serving in the thesis committee. They gave their valuable time and useful comments throughout my master studies.

Moreover, I would like to thank my close colleagues, Dr. Yang Chi Tin, Chiu Kit Chau, Lee Wai Man, Tse Kim Fung, Kwok Ka Wai, Wong Sheung Man, Fu Kin Chung, Lam Hiu Man Josh and Chim Ho Ming for their love during the memorable live in The Chinese University of Hong Kong.

Last but not least, I would like to express my warm thanks to my parents for their endless love and support throughout these years.

Table of Content

List of Figure

List of table

Chapter 1
Introduction

1.1 Motivation

Positioning is a technique that has been presented for thousands of years. Almost every creature in the nature needs to know where it is, in order to navigate from one point to another. Human, for example, originally has the ability using pilotage and celestial navigation which relies on recognizing landmarks or celestial objects such as sun and stars to identify where we are.

Due to the development of the sensing technologies, positioning technologies are implemented and utilized in many diverse applications, including civil and military aviation, cruise missiles, submarines, space technology and automobiles. Moreover, positioning is seen as an important feature of mobile telecommunication. With this technique, mobile unit can either gather the information about its position or be localized from elsewhere. This helps the system to achieve many tasks.

Nowadays, the best known and mostly utilized positioning system is the GPS. A GPS receiver receives signals from at least four GPS satellites to calculate its position. Its accuracy is up to a matter of meters. This excellent and globally available system is well suited for many outdoor positioning tasks. However, since the signals are attenuated or even lost in dense urban areas and inside buildings, GPS has its disadvantages.

Therefore, many researchers work in other positioning systems which can be utilized indoor or as a complement of GPS. INS is one of the solutions. INS relies on initial position, velocity, and attitude and thereafter measuring the attitude rates

and accelerations. Its advantage is that it does not rely on external reference. In other words it does not have signal blockage problem. It seems a nice complement with GPS. However, the disadvantage of INS is that the errors would be accumulated and the accuracy deteriorates with time due to integration. In addition, a high accuracy INS is expensive. Therefore, investigation and development of a low cost and high accuracy INS becomes an interesting topic for researchers.

1.2 Background information

1.2.1 Overview of positioning technologies

Positioning technologies can be divided into two main categories, absolute positioning and relative positioning. Absolute positioning means that the calculated position is independent on the previous result. While relative positioning means that the calculated position depends on the previous result. The following briefly introduces different positioning technologies.

I. GPS

A typical example of absolute positioning is GPS ([1]-[3]). GPS is made up of three parts; satellites orbiting the Earth, control and monitoring stations and GPS receivers owned by users.

In 1993, the United State Air Force launched the 24th Navstar satellite into the orbit, completing a network of 24 satellites. There are six orbits which are equally spaced around the equator with four satellites in each orbit. The orbits are nearly circular with radii of approximately 20200 km. This constellation ensures that a user located anywhere on the Earth has a direct line of sight to at least four satellites at any time. Figure 1-1 shows the orbits of the 24 GPS satellites.

Figure 1-1 Orbits of the 24 GPS satellites

The basic principle of position determination of GPS is time-of-arrival ranging. The GPS receiver contains an internal clock that is utilized to note the time of arrival of the satellite ranging code. Since the satellite's signal is traveled at speed of light, the distance from the receiver to the satellite can be calculated.

As the positions of the satellites are known, together with the distance from the receiver to at least four satellites, the position of the receiver can be obtained. Detail of the calculation can be found in [3].

GPS has evolved from dedicated military systems into true dual-use systems. Its applications can be found in numerous aspects such as recreation and sports, science, transportation, industry, military and spacecraft navigation. Actually, it can accomplish many tasks in outdoor environment.

There exits systems that similar to GPS. One is European Satellite Navigation System, GALILEO [4]. It is launched by the European Union and the European Space Agency. This system will ensure complementarity with the GPS system. Other similar systems are Global Navigation Satellite System (GLONASS) of

Russian, COMPASS navigation system of China and the Indian Regional Navigation Satellite System (IRNSS) of India.

II. Local Position Measurement System (LPM)

Another example of absolute positioning technique is LPM. It is a concept of a kind of wireless sensor network system which consists of at least two separate hardware components. One is a measuring unit that usually collects data and calculates the position. The other is a signal transmitter.

In [6], the concept and overview of the wireless LPM are introduced. LPM can be divided into two categories, self-positioning and remote-positioning system. In self-positioning system, as shown in Figure 1-2, the measuring unit is mobile. It receives the signals of several transmitters in known location and then calculates its current position. Remote-positioning, as shown in Figure 1-3, works in reverse way. The signal transmitter is on the mobile unit and several fixed measurement units receive the signal. The results are collected by a matter station and the position is calculated.

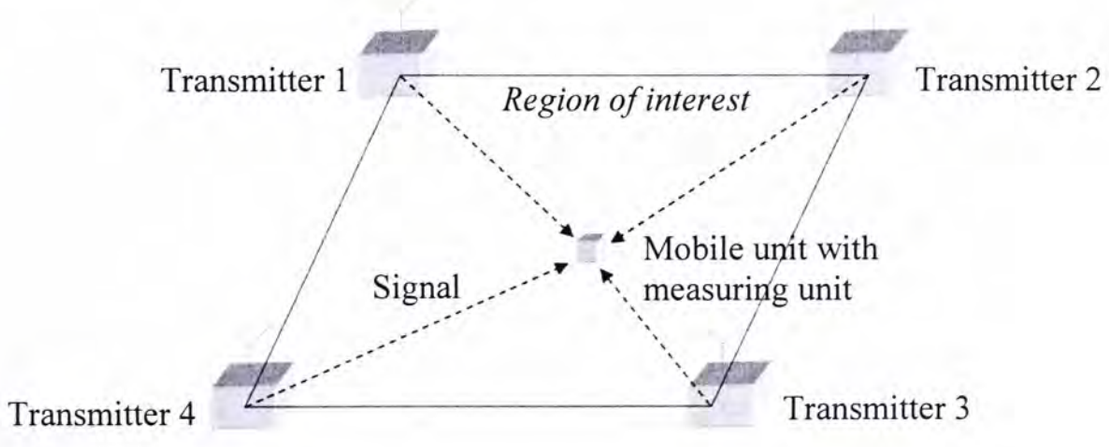


Figure 1-2 Basic construction of self-positioning system

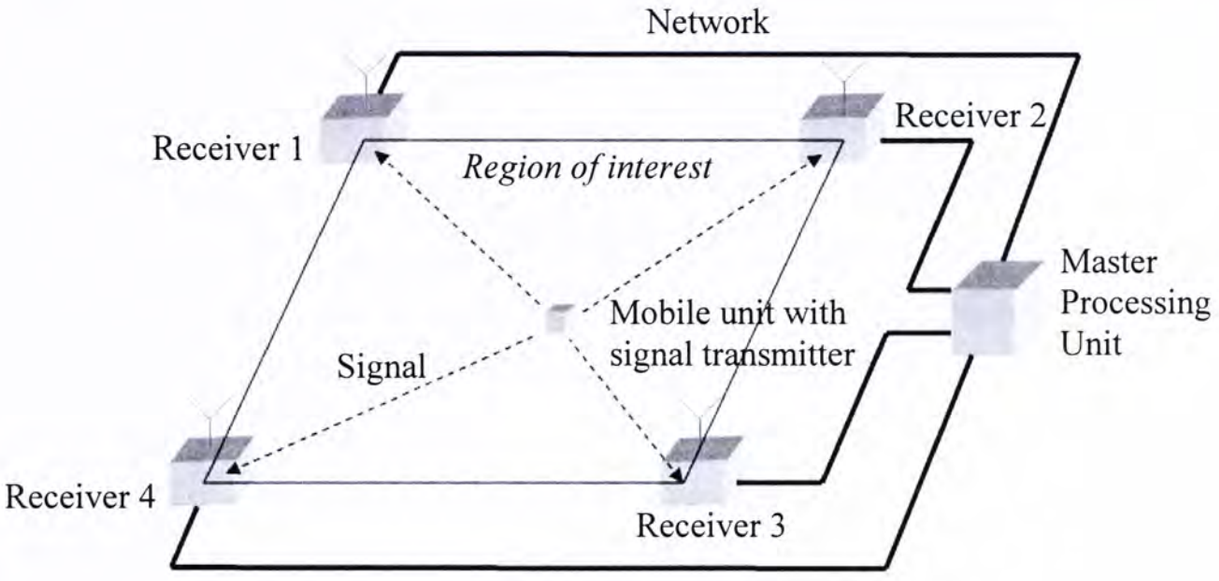


Figure 1-3 Basic construction of remote-positioning system

Similar to GPS, the distance between the signal transmitter and receiver should be determined in order to calculate the position of mobile unit. There are mainly three principles to determine the distance. They are Angle-Of-Arrival (AOA), Received Signal Strength (RSS), and propagation-time based systems that can further be divided into three subclass: Time-Of-Arrival (TOA), Round-Trip-Time-OF-Flight (RTOF) and Time-Difference-Of-Arrival (TDOA). The detail is presented in [6].

Because of the growth of wireless communication technology, the demand of wireless positioning measuring system increases recently [6]. Based on the concept introduced above, many researchers proposed a number of different systems utilizing infrared (IR) [9], ultrasonic [10], radio frequency (RF) ([7], [11]), radio frequency identification (RFID) tags [12], wireless local area network (WLAN) ([13]-[16]) and Bluetooth ([17]-[19]).

III. Dead Reckoning (DR)

Dead (or 'Deduced') Reckoning is an example of relative positioning technology which has been important throughout history. It has been utilized for centuries in marine application and is the method many early aviators utilized to complete early record-setting long-distance flights [3]. The minimum sensing requirements are a direction indicator and a speed indicator. For example, the sailors of early voyage determine the heading direction with compass, while obtain the speed of the vessel by measuring the time of an object travel a fixed length along the side of ship. They combine these and the last known position to figure out where the current position is. Obviously, the calculated position depends on the previous result.

In a modern approach to dead reckoning, the velocity and heading direction of the mobile unit are measured electronically. Since the measured values are based on the moving frame, the actual velocities on the inertial frame can be obtained by transformation that related to the angle of direction. Then the positions on the inertial frame are then obtained by integrating the corresponding velocities with known initial positions and velocities.

IV. INS

A well known relative positioning technique is INS. It is based on the application of Newton's laws of motion. In detail, Newton's first law states that a body in motion tends to maintain constant translational and rotational velocity, unless disturbed by forces or torques. If a force sensing device with a known mass is attached on the body, the acceleration of the body can be determined by applying the Newton's second law.

$$a = \frac{F}{m}$$

The inertial sensors measuring the vector-valued rotation rate and linear acceleration are usually gyroscope and accelerometer respectively.

The basic principle of calculating the current position in INS is as follow. Firstly, transform the linear acceleration and rotation rate from moving frame to inertial frame. Then a single integration yields inertial frame velocity. Finally a second integration provides inertial frame position. As the principle of INS is similar to DR, INS is simply a form of DR [22].

An INS mainly consists of two parts, an inertial measurement unit (IMU) and navigation computers. IMU contains a cluster of inertial sensors, generally accelerometers and gyroscopes. The navigation computer doubly integrates the net acceleration to maintain an estimate of the position of the host vehicle.

There are two main approaches of INS, gimbaled approach and strapdown approach.

1.2.2 Comparison between different positioning systems

Every positioning system has its advantages and disadvantages. Noon of the systems can satisfy all the demands. In this section, the features of difference positioning systems are discussed. As both INS and DR have similar properties, the discussion mainly focuses on GPS, LPM and INS.

GPS is the best well known and most widely spread positioning system. It is globally available. In conventional grade GPS, its accuracy can be up to meters [6]. However, as its principle requires a direct line of sight between the receiver to at least four satellites, signal blockage problem exists. It is not possible to work in dense urban area and indoor environment. Moreover, its accuracy is not high enough for the application needed precise position.

LPM can be made by different wireless communication technologies. So, the performance depends on which technology is utilized. In recent researches, the

accuracy of LPM can achieve the order of cent-meter ([7], [9], [10]). But it still has the problem of signal blockage. The signal would be attenuated by other objects that between the receivers and transmitters. On the other hand, the cover range of the LPM is limited. The system should be installed first in the area of interest. That leads the inflexibility of the system and requires the cost for installation.

INS is a self-contained system. The position found only depends on the data of IMU which is on the mobile unit and the previous result. In other words, it requires no external signal. Therefore, it does not have the signal blockage problem. The main drawback of such relative positioning system is the integration drift. Since the process of determining position requires several integrations, a small error in accelerations can lead a considerable position error. Moreover, as the current result depends on the previous one, the error accumulates. So INS is not a solution of long-term position determination [27].

The properties of these positioning systems are listed below.

	Advantages	Disadvantages
GPS	● Globally available ● Relatively cheap	● Signal blockage problem ● Accuracy is only in order of meters in conventional system
LPM	● Accuracy can achieve the order of cent-meter	● Signal blockage problem ● Cover range of the LPM is limited, not flexible ● Requires the cost for installation
INS	● No signal blockage problem ● Reliable in short-term	● Integration drift ● High cost and bulky size of high grade sensors ● Not a solution of long-term

Table 1-1 Advantages and disadvantages of GPS, LPM and INS

It is obvious that INS contains some features that can solve the problems of both GPS and LPM. Therefore, there are growing research interest in INS recently. The trend of researches related to INS will be discussed in the next section.

1.2.3 Recent works related to INS

INS is a potential solution to solve the weakness of GPS and LPM. It is also a possible method to fulfill the needs of short-term positioning. However, its drawbacks, as listed in Table 1-1, restrict the development of the INS. Therefore, many researches are carried out to tackle these drawbacks. These researches can be mainly divided into three aspects: developing aided INS, reducing the cost and increasing the accuracy.

I. Development of aided INS

In INS, the error exhibited in the acceleration signal is accumulated and the accuracy of the distance measurement deteriorates with time due to the integrations. This problem can be fixed by periodic recalibration by the external measurements on position, velocity, and attitude. The INS assisted by other sensing measurements is called "aided INS".

A common external signal source is GPS. It is a reliable technique when the signals are clear. There are numerous researches on the integration of GPS and INS ([24]-[27]). A common method to integrate two systems is utilizing Kalman Filter [3]. Besides, direct Kalman filter approach [24], adaptive Kalman filter approach ([25], [26]), neural networks [27] are the methods proposed to solve the problem of data fusion.

Apart from GPS, other sensing devices such as vision ([32], [33]), magnetometer [34] and optical encoder [35] are utilized to provide reference to INS.

II. Cost Reduction

Tradition inertial sensors such as accelerometers and gyroscopes are high cost, large weight and bulky size. Because of the development of MEMS, the cost, weight and size of inertial sensors can be greatly reduced. Therefore, there is increasing number of researches utilizing MEMS based INS ([20], [21], [25], [32]-[35], [38]-[41]).

Comparing the MEMS accelerometer and MEMS gyroscope, MEMS accelerometer is more affordable in the same level of precision [45]. On the other hand, MEMS gyroscope is generally more inferior in terms of noise, accuracy, linearity and mechanical durability due to the inherent design difficulties in miniaturization [44]. So replacing gyroscopes by accelerometers to form a Gyroscope-Free INS (GF-INS) is the focus of some researches recently ([42]-[46]).

III. Enhancing the accuracy

One challenge of utilizing low cost MEMS inertial sensors is to get rid of the decrease of accuracy compared to the tradition inertial sensors. Some researches ([39], [40]) work on increasing the reliability of MEMS inertial sensors. Methods like Kalman filter [41] and fuzzy logic [38] are also proposed. Furthermore, the error analysis and modeling of the whole system are investigated ([29], [30]). Novel modeling method utilizing quaternions is also discussed ([36], [37]).

1.3 Objective

The objective of the project is to build a system that can determine the position of mobile unit. After investigating different types of positioning technologies, we decided to build a positioning system based on INS. In [22], A. D. King gave a review of development of inertial navigation and made a forecast: "*integrated (multi-source) navigation systems will continue to increase in sophistication, and will continue to require inertial data as an essential ingredient*". The trend is that INS will still take an important role in positioning technique in the near future. A low cost, high accuracy INS will benefit the integrated navigation system. Hence, the aims of this project are:

- Study the principle of INS and GF-INS
- Investigate the performance of MEMS accelerometer in position determination
- Construct INS and GF-INS to determine the position of mobile unit based on MEMS accelerometer
- Investigate the error of the system
- Demonstrate methodology to enhance the accuracy

1.4 Organization of thesis

This thesis is divided into seven chapters. Chapter one presents the introduction of this project. The overview of different positioning technologies and the objective of this project are also included. Chapter two is the literature study. The basic knowledge and the algorithm related to both INS and GF-INS are introduced.

Chapter three gives the investigation and evaluation of the performance of MEMS accelerometer in position determination. A one dimensional INS is constructed as a testing setup. MEMS accelerometer ADXL103 form Analog Devices Inc. is utilized. Initial calibration and filtering methods are included.

Two noise reduction methods to improve the performance of the one dimensional INS are proposed in chapter four. The first one is a steady state detector based on fuzzy logic. Another is utilizing Kalman filter. Experimental results illustrate the performance of the methods in noise reducing in both moving and steady state.

Chapter five gives an introduction on algorithm of GF-INS. Moreover, a GF-INS with six MEMS accelerometers is constructed. The setup is tested with the robot arm platform in both linear and angular motions.

In chapter six, two methods to enhance the performance of the GF-INS in position determination are proposed. The first one is the modified configuration error compensation. It reduces the effect of scale factor error, bias error, sensing direction error and the position error. The second one is a fuzzy rule based motion state detector. Gradient descent is proposed to train the fuzzy membership functions. Experimental results are also discussed.

Finally, the conclusions of this thesis are given in chapter seven. Furthermore, commendation for the future works is also given.

Chapter 2
Literature Study

2.1 Introduction to INS

The basic principle of INS is fundamental physics. By integrating the accelerations we got from INS, velocities are obtained. Displacements are determined by integrating the velocities. As the sensed data and the displacement that we are interested in are represented in difference coordinate frames, transformation technique between the two frames is needed. Base on this principle, two main types of INS are developed. They are gimbaled INS and strapdown INS.

2.1.1 Coordinate Frames

The process of positioning system is defined relative to a known reference which is usually defined as a specific coordinate system. Sensors are also resolved relative to a particular coordinate system. When these two coordinate systems do not coincide, it is necessary to transform points or vectors between them.

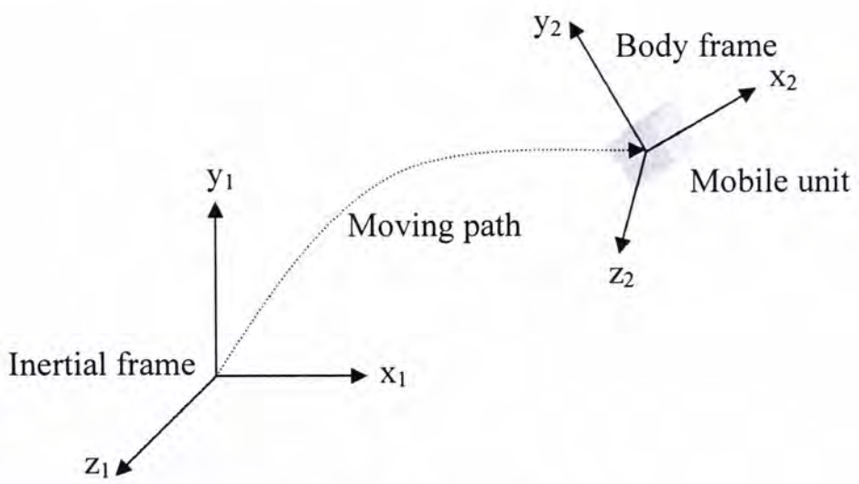


Figure 2-1 Coordinate frames utilized in this project

For simplicity, two coordinate frames are defined in this project, inertial frame and body frame, as shown in Figure 2-1. The origin of the inertial frame is the initial position of the mobile unit. The current position or displacement of the mobile unit after moving is represented in inertial frame. The body frame is rigidly attached to the mobile unit. The measurements of the sensors are based on the body frame. Therefore, transformation is needed to align the measurements to the axes of inertial frame.

One method of transformation is utilizing Euler angles. According to the Euler's rotation theorem, any rotation may be described by three angles. There are several conventions for Euler angles, depending on the axes about which the rotations are carried out. For example, the transformation from body frame to inertial frame can be described by rotation matrix R_b^i, which is formed by roll R_ϕ, pitch R_θ and yaw R_ψ about the x, y and z axes of body frame respectively. The rotation matrixes are given as follow.

$$R_\phi = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \tag{2-1}$$

$$R_\theta = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \tag{2-2}$$

$$R_\psi = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2-3}$$

$$R_b^i = R_\phi \cdot R_\theta \cdot R_\psi \tag{2-4}$$

Note that the rotation matrixes in (2-4) must be multiplied in the corresponding order. When the body frame rotates, the angles can be updated by the measurements of the gyroscopes (ω_x, ω_y and ω_z).

$$\dot{\phi} = (\omega_y \sin\phi + \omega_z \cos\phi)\tan\theta + \omega_z \tag{2-5}$$

$$\dot{\theta} = \omega_y \cos\phi - \omega_z \sin\phi \tag{2-6}$$

$$\dot{\psi} = (\omega_y \sin\phi + \omega_x \cos\phi)\sec\theta \tag{2-7}$$

Simply integrate the angular rates and update R_b^i. Then the measured accelerations in body frame can be transferred to inertial frame by

$$a_i = R_b^i a_b \tag{2-8}$$

Another method is utilizing direction cosine matrix C_b^i. It is defined as the cosine of the angles between the unit vectors in two coordinate frames.

$$C_{xyz}^{uvw} = \begin{bmatrix} \cos(\theta_{xu}) & \cos(\theta_{xv}) & \cos(\theta_{xw}) \\ \cos(\theta_{yu}) & \cos(\theta_{yv}) & \cos(\theta_{yw}) \\ \cos(\theta_{zu}) & \cos(\theta_{zv}) & \cos(\theta_{zw}) \end{bmatrix} \tag{2-9}$$

It is propagated by

$$\dot{C}_b^i = C_b^i \Omega \tag{2-10}$$

Where Ω is a skew-symmetric matrix representing rotation rates which are the measurement from the INS.

$$\Omega = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \tag{2-11}$$

The accelerations measured can be transferred with respect to the body frame.

$$a_i = C_b^i a_b \tag{2-12}$$

Compare to Euler angle approach, direction cosine matrix is more common in navigation system. One reason is that it does not have problem of singularity which Euler angle approach has (equations (2-5) and (2-7)). Moreover, direction cosine matrix is less computationally expensive.

2.1.2 Gimbaled INS

Gimbaled INS is the first type of INS proposed around 1940s. It contains at least three gimbals to isolate a subsystem from host mobile. A gimbal is a rigid frame with rotation bearings. In ideal case, the bearings are frictionless and the fame could be perfectly balanced, then the rotational inertial of the frame would be sufficient to isolate it from rotations of the supporting body. However, gyroscopes are utilized in real case to detect any rotation of the frame due to bearing friction or frame unbalance. This design ensures the innermost IMU maintains its orientation in inertial frame. In other words, the inertial frame accelerations can be measured directly from the accelerometers. The gimbaled INS is showed in Figure 2-2.

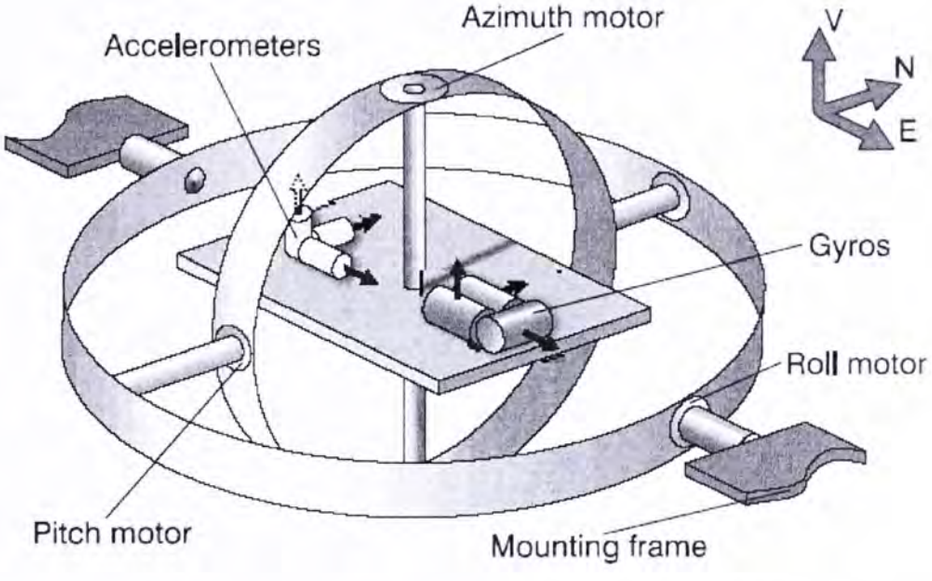


Figure 2-2 Construction of gimbaled INS (from reference [22])

Gimbaled INS can be very reliable and accurate. Its application can be found in aircraft, space launchers, missiles and land vehicles in early age [22]. However, the gimbal arrangement is mechanically very complex. It is also expensive to manufacture and maintain. So from the early 1970s, researchers started contemplating an alternative and simpler arrangement. That is the approach of strapdown INS.

2.1.3 Strapdown INS

A strapdown INS is a major hardware simplification of the gimbaled INS. The gyroscopes and accelerometers are hard mounted ("strapped down") to a common base. The base is not able to mechanically moved and no longer inertially stabilized. The gyroscopes in this system are not used to maintain the accelerometer axes stabilized. Instead, a software solution is utilized to keep track of the orientation of the IMU and rotate the measurements from body frame to inertial frame. This method reduces the mechanical complexity of gimbaled INS and most importantly reduces the cost, size and power consumption. However, the inertial sensors for strapdown systems experience much higher rotation rates than their gimbaled counterparts. Attitude rate compensation for accelerometers is needed to reduce the rotation induced errors such as the centrifugal accelerations in rotating accelerometers.

2.1.4 Conventional algorithm of strapdown INS

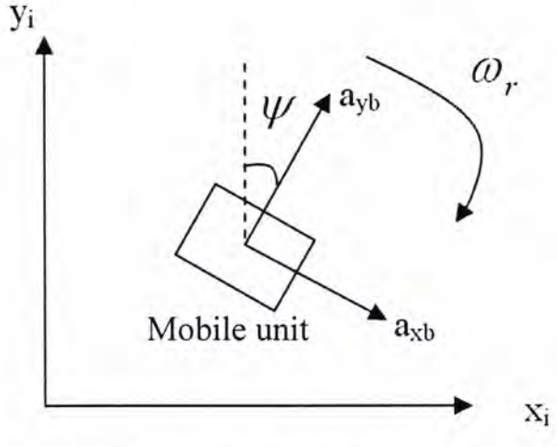


Figure 2-3 Ideal two dimensional strapdown INS

Figure 2-3 shows an ideal two dimensional INS. Two linear accelerometers are utilized to sense the linear accelerations (a_{xb} and a_{yb}) in body frame, while a gyroscope is utilized to determine the rotation rate (ω_r).

The basic principle of calculating the current position in INS is as follow. Firstly, transform the linear accelerations and rotation rate from body frame to inertial frame. Then a single integration yields inertial frame velocities. Finally a second integration provides inertial frame position.

In this example, the body frame accelerations are transformed to inertial frame by this,

$$\begin{bmatrix} a_{xi} \\ a_{yi} \end{bmatrix} = \begin{bmatrix} \cos(\psi) & -\sin(\psi) \\ \sin(\psi) & \cos(\psi) \end{bmatrix} \begin{bmatrix} a_{xb} \\ a_{yb} \end{bmatrix} \tag{2-13}$$

$$\text{where } \psi = \int_o^t \omega_r(\tau) d\tau$$

Then the positions with respect to inertial frame are

$$p(t) = \int_0^t \int_0^\tau \begin{bmatrix} a_{xi}(s) \\ a_{yi}(s) \end{bmatrix} ds d\tau \tag{2-14}$$

Therefore, with a known starting point and initial conditions, INS can obtain the position of the mobile unit according to its linear accelerations and rotation rate. As the principle of INS is similar to DR, INS is simply a form of DR [22].

In three dimensional case, conventional design is utilizing three accelerometers and three gyroscopes. Sensors are mounted on a rigid body attached to the mobile unit. Then the sensed angular velocities and linear accelerations are with respect to the body frame. The process is similar to the two dimensional case. The procedures of signal processing in conventional strapdown INS are described in the block diagram in Figure 2-4.

There exists GF-INS utilizing only accelerometers ([42]-[46]). This project will concentrate on this kind of system. The detail will be discussed in chapter five.

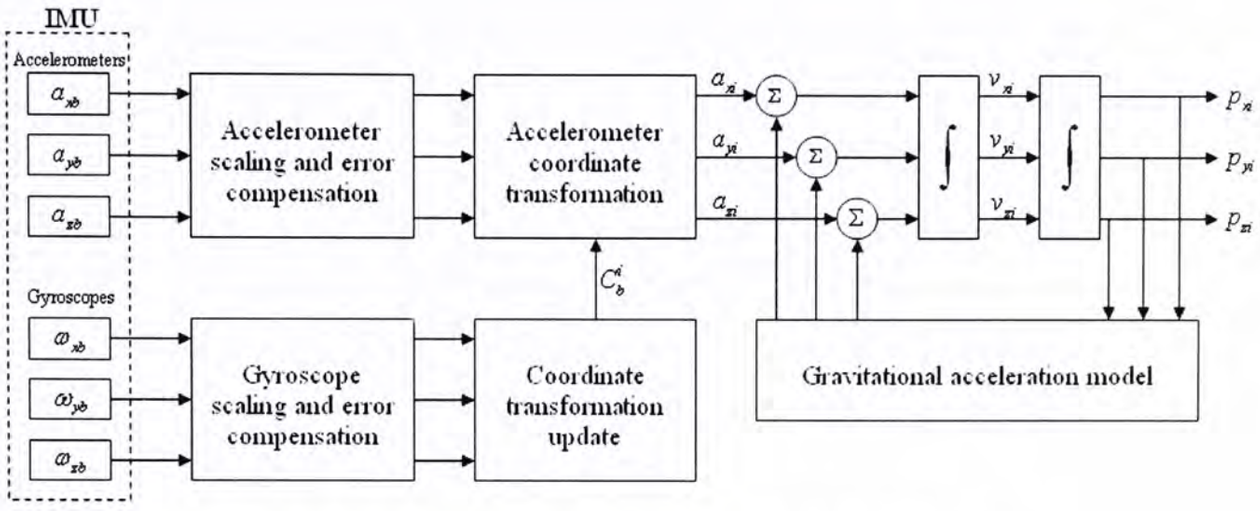


Figure 2-4 The block diagram of conventional strapdown INS

2.2 Inertial sensors

Gyroscope and accelerometer are two types of sensors commonly utilized in inertial sensing. Because of the low cost and small size, MEMS based sensors are considered in this project. Brief introduction of MEMS gyroscope and accelerometer are presented in this section.

2.2.1 Gyroscope

Gyroscope is a device to sense the rotation rate. Most of the MEMS gyroscopes utilize vibrating mechanical elements to sense rotation [47]. The operating principle is based on the Coriolis acceleration. It is an apparent acceleration that arises in a rotating reference frame and is proportional to the rate of rotation.

$$a_{Coriolis} = -2\omega \times v \tag{2-15}$$

Where v is the velocity of an element in the rotating system and ω is the rotation rate along the rotation axis of the reference frame.

There are several implementation methods of MEMS gyroscope. For example, the gyroscopes from Analog Devices Inc. contain two polysilicon sensing structures with dither frames which are electrostatically driven to resonance. The necessary velocity element is then provided. Capacitive pickoff structures at the outer extremes of each frame are utilized to sense the Coriolis acceleration. If the sensed body rotates, the element experiences Coriolis Effect. With the velocity and the Coriolis acceleration, the rotation rate can be found. The output of the gyroscope is proportional to the rotation rate.

2.2.2 Accelerometer

Accelerometer is a device to sense the linear acceleration. It is generally a spring mass damper system which consists of a proof mass suspended by compliant beams anchored to a fixed frame.

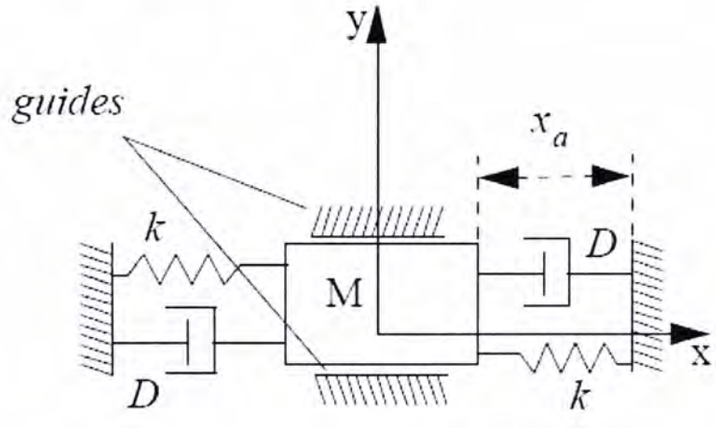


Figure 2-5 General structure of accelerometer

Figure 2-5 shows that simple structure of accelerometer. By Newton's laws, the dynamic equation of the system can be represented by

$$\ddot{x}_a = -\frac{k}{m}x_a - \frac{D}{m}\dot{x}_a \tag{2-16}$$

Since the dynamic response of modern MEMS accelerometer exceeds most rigid body dynamics by at least an order of magnitude [45], the damping D can be

neglected. Therefore, if the moving distance x_a is found, the corresponding acceleration can be obtained.

In this project, MEMS accelerometer ADXL103 from Analog Devices Inc. is utilized. The basic principle to relate the motion is utilizing the capacitance variation. When the object accelerates, the mass in the sensor moves such that the capacitances between the plates vary, as shown in Figure 2-6. The change of the capacitance is defined by

$$C = \varepsilon \frac{A}{x_a} \tag{2-17}$$

Where ε is the permittivity of the air gap and A is the surface area of the plate. By measuring the change of capacitances, the output of the accelerometer is proportional to the acceleration applied to the object.

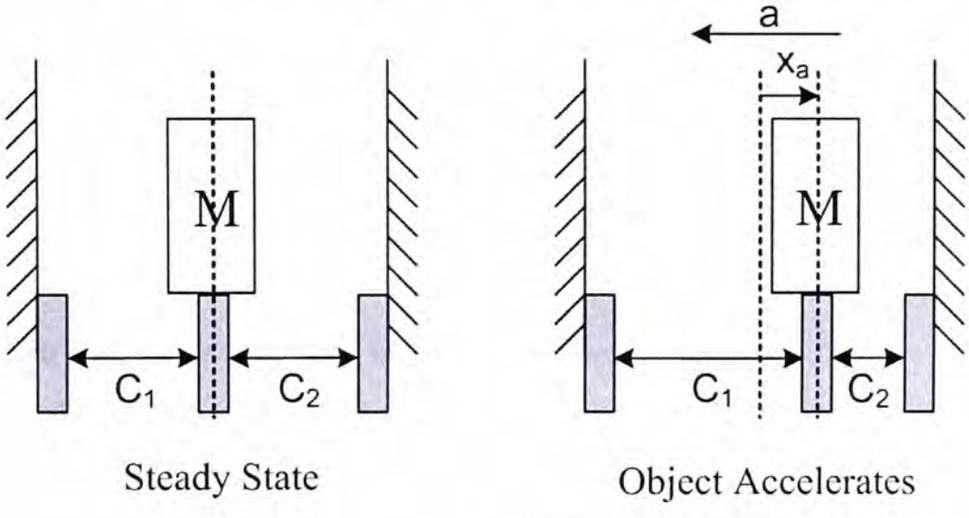


Figure 2-6 Basic principle of accelerometer

Compare to MEMS gyroscope, MEMS accelerometer is low cost and small size. In the conventional grade, accelerometer with multi-axes costs several times lower than single axis gyroscope [48]. Moreover, design and fabrication advancements in MEMS accelerometer are fast exceeding those of gyroscope ([44], [47]). That leads accelerometer is more reliable than gyroscope. Some researches ([42]-[46]) state that a network of accelerometers can replace

gyroscopes to form GF-INS. This project follows and extends the work of GF-INS. Therefore, only MEMS accelerometers are utilized.

2.3 Previous works

There are some researches ([20], [21], [25], [32]-[35], [38]-[41]) related to MEMS based INS. As discussed in section 1.2.3, aided INS is a trend of the researches. In ([20], [25], [34]), low cost INS aided by GPS are discussed. An aided INS based on inertial, air data and magnetic sensor is presented in [21]. Vision data is also a possible data to aid INS. Aided INS utilizing a multi-dimensional stochastic feature tracker is introduced in [32]. An indoor personal positioning system based on wearable INS and camera is demonstrated in [33]. Utilizing optical encoders on the wheel of mobile robot is also a method providing source to calibrate INS. [35] shows such system.

Periodically calibrating INS is a method to increase the accuracy. Another method is reducing the error of INS alone. An evaluation of low cost MEMS accelerometer is described in [41]. The result shows that the accelerometer can be a solution for short duration distance measurement. A Kalman filter is utilized to reduce the random noise of the sensor. It also shows experiment to find the random bias drift of the accelerometer. The performance of the accelerometer is shown to be acceptable.

Park and Gao [39] present the error analysis and stochastic modeling of the low cost MEMS accelerometer. They state that it is significantly important to characterize the error behaviors of the sensors and have a suitable stochastic model. The presented model can improve the performance of accelerometer and help the development of optimal algorithm for integration of MEMS accelerometers and other systems.

Although the aim in [40] is developing an error reduction strategy for measuring the displacement of parallel kinematic machine struts and the tool centre point pose, it is a good reference since it includes error analysis and error reduction for MEMS accelerometer. The model presented in [39] is adopted and methods to identify the errors are shown.

A fuzzy logic based method to reduce the error in MEMS accelerometer is demonstrated in [38]. A non-linear fuzzy logic filter has been implemented with MEMS accelerometer ADXL103. The results show improvement of the MEMS sensor accuracy. This paper introduces a new insight in applying fuzzy logic in error reduction in signal processing of accelerometer.

2.4 GF-INS

Typically, INS uses accelerometers to sense linear accelerations and gyroscopes to sense the angular velocities. As stated in section 1.2.3, there is motivation to replace the gyroscopes by accelerometers in traditional INS.

Using linear accelerometers to measure a moving vehicle was first proposed by Schuler in 1967 [49]. He proposed a moving vehicle analysis method requested at least nine accelerometers. In 1994, Chen [50] proposed a method to compute both angular and linear motion by utilizing only six accelerometers. Recently, Tan and Park [45] found that a minimum of six accelerometers is required for a complete description of rigid body motion.

Both linear and angular accelerations can be sensed by a network of accelerometers which are in a special cube-shape configuration. Each accelerometer is placed in the centre of the diagonal in each side of the cube. Figure 2-7 shows the configuration of the six accelerometers. Each side of the

cube is 2L long. The arrows represent the sensing axes of the accelerometers. The body coordinate frame can be defined in the centre of the cube.

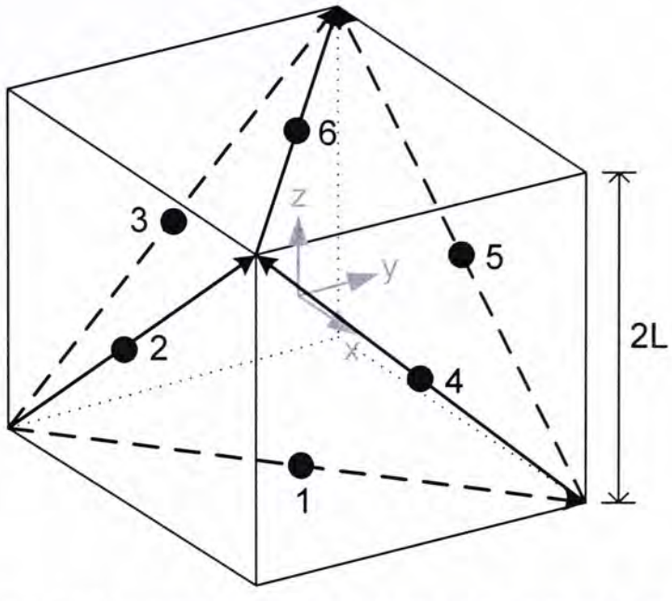


Figure 2-7 Configuration of GF-INS from Tan and Park [45]

With this cube-shape configuration, the dynamical system of equations of the target body is simplified since the angular accelerations become linear combination of the accelerometer outputs.

$$\begin{bmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\omega}_3 \end{bmatrix} = \frac{1}{2\sqrt{2}L} \begin{bmatrix} A_1 - A_2 + A_5 - A_6 \\ -A_1 + A_3 - A_4 - A_6 \\ A_2 - A_3 - A_4 + A_5 \end{bmatrix} \tag{2-18}$$

$$P = \frac{1}{2\sqrt{2}} \begin{bmatrix} A_1 + A_2 - A_5 - A_6 \\ A_1 + A_3 - A_4 + A_6 \\ A_2 + A_3 + A_4 + A_5 \end{bmatrix} + L \begin{bmatrix} \omega_2\omega_3 \\ \omega_1\omega_3 \\ \omega_1\omega_2 \end{bmatrix} \tag{2-19}$$

Where $\dot{\omega}$ are the angular accelerations while P describes the linear accelerations of the target body.

This system of equations only works with the ideal cube-shape configuration. So Tan and Park also give methods to identify the configuration errors such as

accelerometer location and orientation errors. A compensator is also presented to reduce such errors.

In this project, GF-INS algorithm is studied. An experimental setup is developed based on this algorithm and MEMS accelerometers. The detail will be given in chapter five.

2.5 Summary

In this chapter, some basic knowledge related to this project is introduced. Two coordinate frames, inertial frame and body frame, utilized in this project are defined. There are three common methods to transform the reference frame of the sensed data to another. They are Euler angles, direction cosine matrix and quaternion. The first two are discussed. It is found that direction cosine matrix is a more common method utilized in navigation system.

Then two main implementation methods of INS, gimbaled and strapdown INS are shown. Gimbaled INS is a very reliable and accurate system. However, its mechanical complexity and the requirement on the high precision inertial sensors are the main disadvantages. Strapdown INS is mechanically simpler and becomes the trend of nowadays INS. The principle of the strapdown INS is studied. The system developed in this project is based on the concept of strapdown INS.

The inertial sensors utilized in INS are mainly gyroscope and accelerometer. The working principles of both sensors are discussed. In this project, MEMS accelerometer is investigated and utilized to build an INS.

MEMS based INS is still an interesting topic for researches. Some related works are presented. In order to increase the accuracy of MEMS based INS, some focus on aided INS. The idea is utilizing external reference source to calibrate the INS,

resulting reduce the effect of integration drift. Others are working on modeling the errors including configuration and stochastic errors of MEMS base INS alone.

Finally, the works related to GF-INS are presented. The recent results of Tan and Park in GF-INS are included. They found that a set of six accelerometers in cube-shape configuration is enough to sense the angular and linear motion of the target body. Base on this result and the investigation of MEMS accelerometers, a setup of GF-INS is developed. The details will be discussed in next few chapters.

Chapter 3
Performance of MEMS accelerometer in position determination

The Objective of this project is to build a low cost high accuracy INS based on MEMS accelerometer. We start the work by investigating and evaluating the performance of MEMS accelerometer in position determination. To do so, a one dimensional INS with one MEMS accelerometer is developed. Similar work can be found in [41]. A dual axes accelerometer ADXL202 is utilized. While in this project, a single axis ADXL103 from Analog Devices Inc. is investigated. In this chapter, the details of the system are shown. Initial calibration and noise filtering are proposed to enhance the system performance.

3.1 Basic principle

In one dimensional case, there is no rotation and no need for gyroscope. Only one accelerometer is sufficient. Suppose the initial position and velocity of the mobile unit are known. By utilizing accelerometer, the acceleration of the mobile unit is sensed and recorded in a period of time. Based on the fundamental physics, doubly integrating the acceleration with known initial conditions gives position moved.

Figure 3-1 shows the working flow of the system. The sensing axis of the accelerometer is set the same as x-axis of body frame.

Since there is no rotation, the transformation between frames can be omitted by choosing the inertial and body frames in the same attitude. Moreover, gravitational acceleration can be neglected if the sensing direction of the accelerometer is set at perpendicular to the gravity.

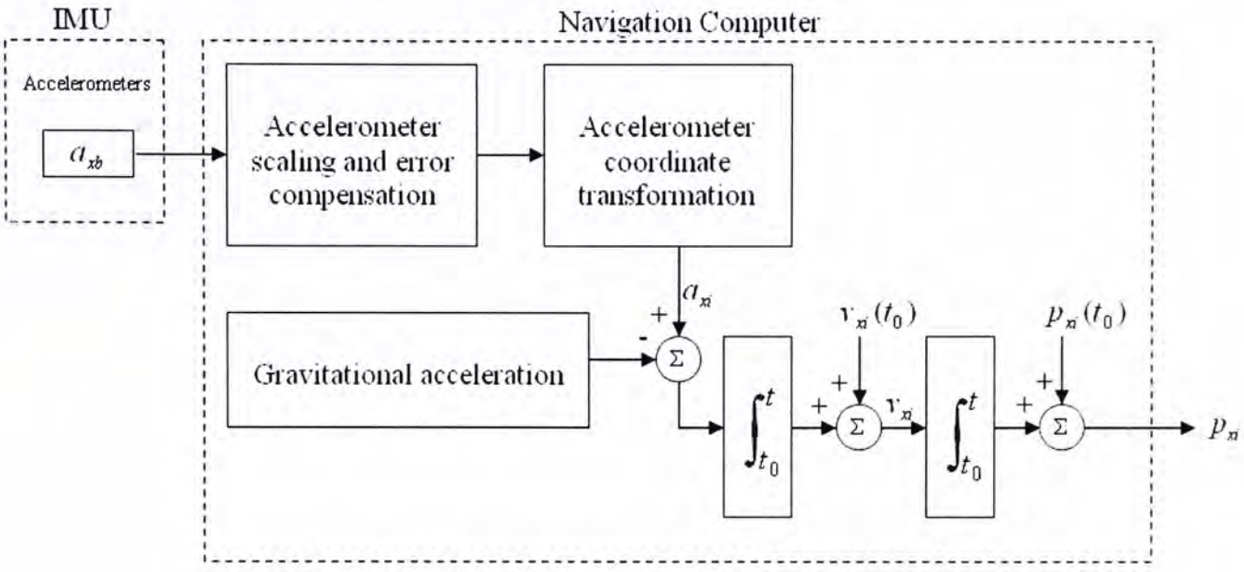


Figure 3-1 Principle of one dimensional INS

3.2 Numeric integration

In order to find the velocity and displacement from acceleration, numeric integration is needed because the sensed data is a sequence of discrete data points with certain time interval. In continuous case, the velocity is the area under acceleration curve in the acceleration-time graph. The same concept is applied in discrete case. A common method is Euler integration.

By the definition of the derivative,

$$\dot{v}(t) = \lim_{\Delta t \to 0} \frac{v(t+\Delta t) - v(t)}{\Delta t} \tag{3-1}$$

If Δt is very small,

$$\frac{v(t+\Delta t) - v(t)}{\Delta t} \approx \dot{v}(t) = a(t) \tag{3-2}$$

Therefore

$$v(t+\Delta t) \approx a(t)\Delta t + v(t) \tag{3-3}$$

If the initial condition $v(t_0)$ is known and the time step Δt is small, the corresponding velocity can be found. Similarly, displacement can be determined when initial condition $x(t_0)$ is also known. The graphical representation is shown in Figure 3-2. The sum of shaded region is the calculated velocity.

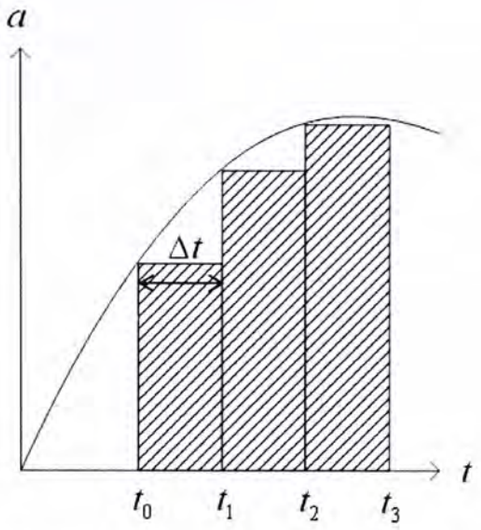


Figure 3-2 Graphical representation of Euler integration

In this project, trapezoidal rule is utilized as the integration method. Its principle is similar to the Euler integration. Instead of counting the area of rectangle between the time intervals, area of trapezoid is taking in this method. The graphical representation is shown in Figure 3-3.

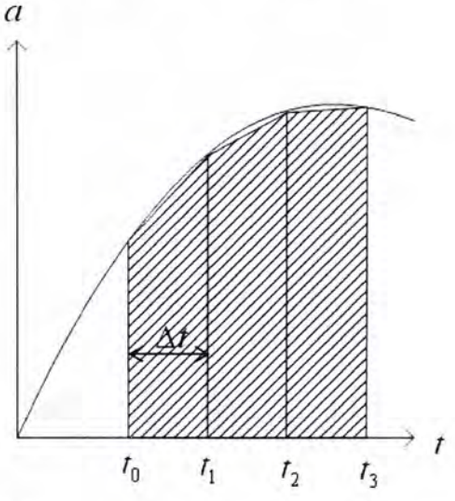


Figure 3-3 Graphical representation of trapezoidal rule

The velocity is nearly the same as the area of the trapezoids when the time step Δt is very small.

$$v(t+\Delta t) \approx (a(t)+a(t+\Delta t))\frac{\Delta t}{2}+v(t) \tag{3-4}$$

3.3 Experimental setup

In order to have hands-on experience and evaluate the MEMS accelerometer in position determination, a one dimensional INS is constructed. The system contains one accelerometer and microcontroller. The data is recorded in frequency 100 Hz. The whole setup can be controlled by a master computer wirelessly. In this stage, the sensed data is transferred to the master computer for further analysis. On the other hand, the system can be self-contained because it contains a microprocessor to calculate the position traveled.

3.3.1 MEMS Accelerometer

The accelerometer utilized in this project is ADXL103 from Analog Devices Inc.. It is a high precision, low power, complete single-axis accelerometers with signal conditioned voltage outputs. The full-scale range of $\pm 1.7g$ can be measured. The working principle of this sensor is similar to that described in section 2.2.2. The sensed acceleration is proportional to the output voltage.

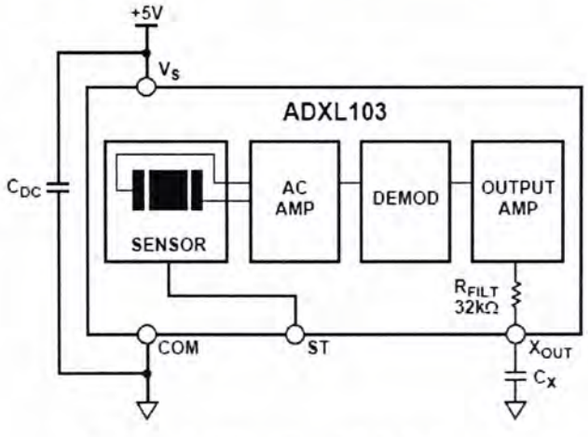


Figure 3-4 Functional block diagram

Figure 3-4 shows the functional block diagram of the sensor. According to the data sheet [51], the two capacitors, C_x and C_{DC}, are set to $0.1\mu F$ such that the bandwidth of the accelerometer is $50Hz$. A 100Ω resistor is added to reduce the noise from the supply. The signal in X_{out} is analog voltage and proportional to the acceleration. Figure 3-5 shows the circuit designed for the accelerometer.

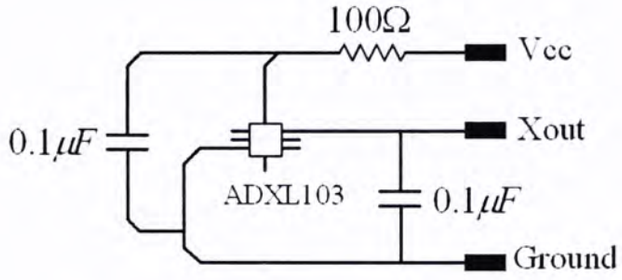


Figure 3-5 Circuit diagram of ADXL103

The features of the accelerometer are presented in Table 3-1. Note that the sensitivity and zero g bias are based on 5V supply. It may slightly different when working in other supply voltage.

Range	$\pm 1.7g \quad (g = 9.80708 m/s^2)$
Power Supply	3-6 V
Size	5 mm × 5 mm × 2 mm
Bandwidth	50 Hz
Resolution	1 mg
Sensitivity	1000 mV/g
Zero g bias	2.5 V ± 0.025 V

Table 3-1 Features of ADXL103

A sensor circuit is then constructed. The output of the sensor is connected to an analog-to-digital converter in which the signal is sampled in 100Hz. The figure below illustrates the sensor board developed. The bottom side contains a 51-pin connector for connecting the microcontroller.

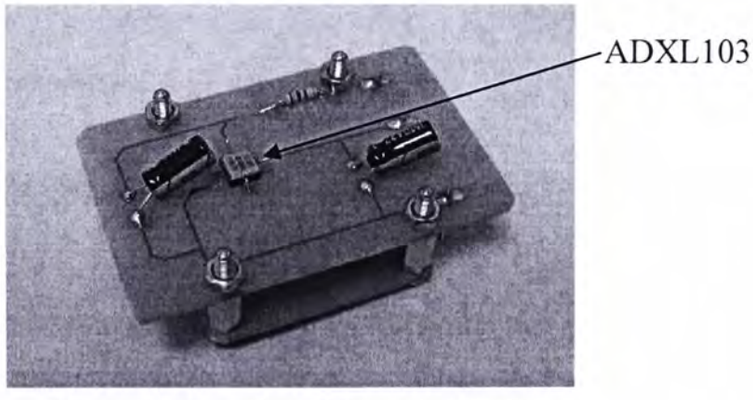


Figure 3-6 Overview of the sensor board

3.3.2 Microcontroller

MICA Mote is utilized as the microcontroller in this project. It is developed by UC Berkeley's research group for research and development of low power, wireless sensor networks. MICA Mote (MPR300CB) consists of Atmega 128L processor which is driven by a software operating system called TinyOS, ISM band radio transceiver module, 51-pin connector for connecting the senor broad, and two AA size batteries for power supply. It is an integrated system with the advantages of small size and easy implement. Figure 3-7 shows the overview of the MICA Mote.

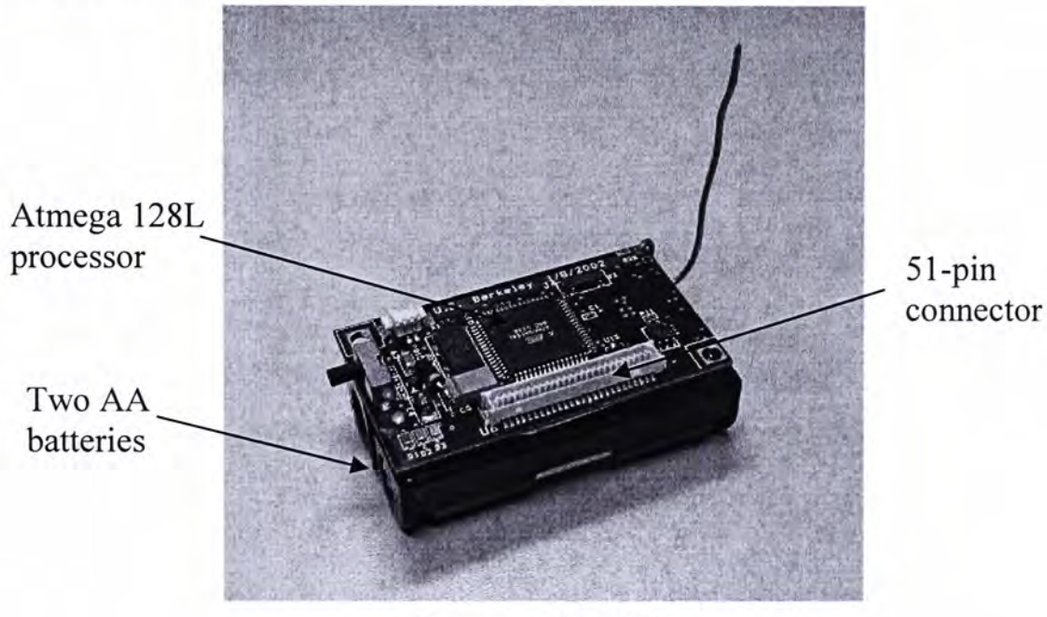


Figure 3-7 MICA Mote

The processor Atmega 128L contains eight channels of 10-bit analog-to-digital converters. The output of the sensors is connected to the input of one of the channels through the 51-pin connector. The sampling frequency is controlled by the processor and the reading is defined as follow,

$$ADC = \frac{V_{in}}{V_{ref}} \times 1024 \tag{3-5}$$

Where V_{in} is the input voltage while V_{ref} is the reference voltage. The MICA Mote also contains a 512 kbyte flash EEPROM which can be utilized to store the user defined program and the sensed data.

TinyOS is an embedded operating system in which programming language "necC" is utilized to program the MICA Mote. To achieve communication between the master computer and the mote, interface board MIB300CA is necessary. Through this interface, command can be sent from master computer to the mote which is connected to the interface, and then to another mote wirelessly. The connection of two MICA Motes, interface broad and the master computer is illustrated in Figure 3-8.

Figure 3-8 Connection of two MICA Motes, MIB300CA and master computer

3.3.3 System architecture

In this system, two MICA Motes are utilized. One is connected to the master computer as a radio communication base station while another is connected to the sensor broad as a controller. Two necC programs to drive the motes are built and downloaded to the motes. Figure 3-9 shows the architecture of the system. The MICA Mote 2 and the sensor broad are mounted on the mobile unit.

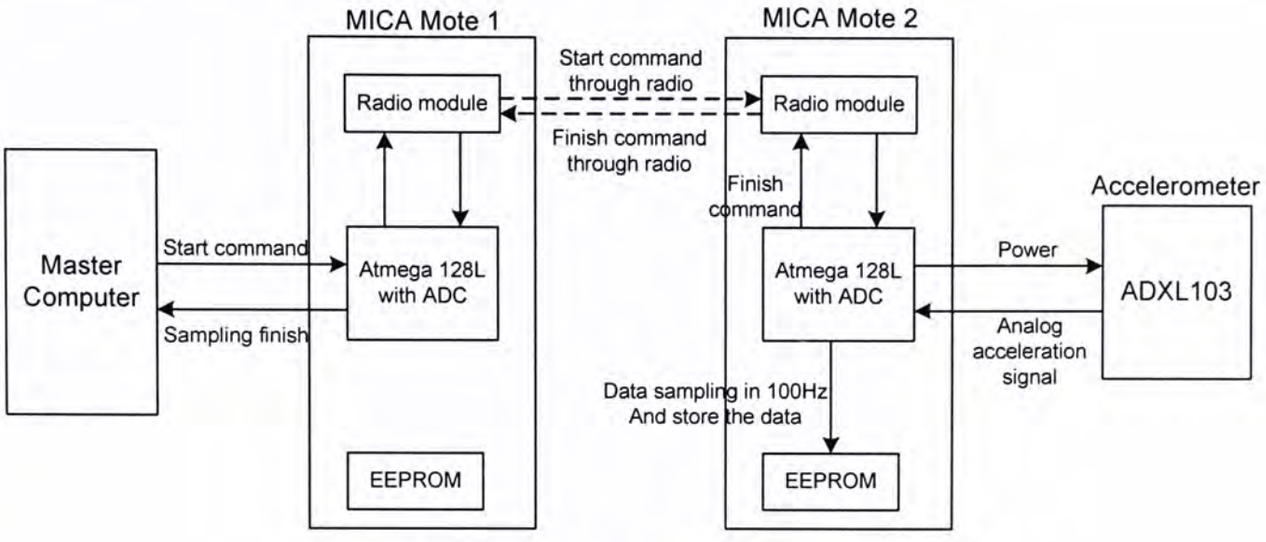


Figure 3-9 Architecture of the system

Start command is sent to the MICA Mote 1 when a JAVA program is run by the user. The command includes the sampling times and quantity of the samples. Then the command is transfer to the MICA Mote 2 through built in radio module. The processor in MICA Mote 2 starts sampling the voltage output of the accelerometer with user defined frequency and amount. The value of the ADC output is defined as equation (3-5). The sensed data are stored in the EEPROM first for further analysis. When the sensing ends, MICA Mote 2 sends signal to the MICA Mote 1 and then the master computer to end the sensing process. The data is extracted by connecting MICA Mote 2 and the master computer. At this stage, the calculation of the distance traveled is doing offline because it is convenience for further analysis of the data. To implement online calculation, the sensed data are sequentially transferred to the MICA Mote 1 and master computer through radio instead of storing in the EEPROM.

3.3.4 Testing platform

In order to test the performance of the INS, a general-purpose industrial robot with five degree of freedom (DOF) manufactured by Eshed Robotec Limited is utilized. The overview of the robot arm is as follow.

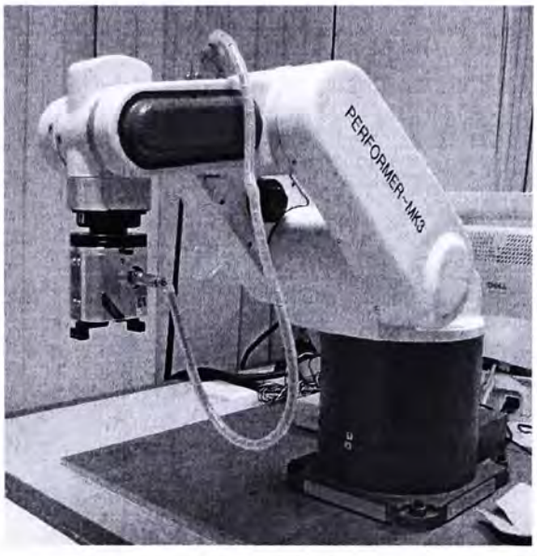


Figure 3-10 Five DOF robot arm

The tool end of the robot acts as a mobile unit which can move in three dimensional space. The moving trajectory is controlled by the user and distance of travel can be obtained. Therefore, the determined distance from INS can be compared with the real trajectory for further analysis. The programming language to control the robot is Advanced Control Language (ACL), which is stored onto a set of EEPROMs. The robot is controlled based on two coordinate systems, joint coordinate system and Cartesian coordinate system. All points in the three dimensional space are represented in these two coordinate systems. The joint coordinate system and Cartesian coordinate system are shown in Figure 3-11 and Figure 3-12 respectively.

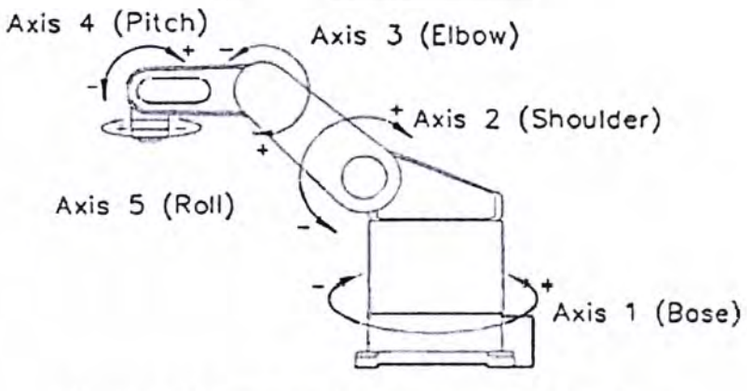


Figure 3-11 Joint coordinate system of the robot arm

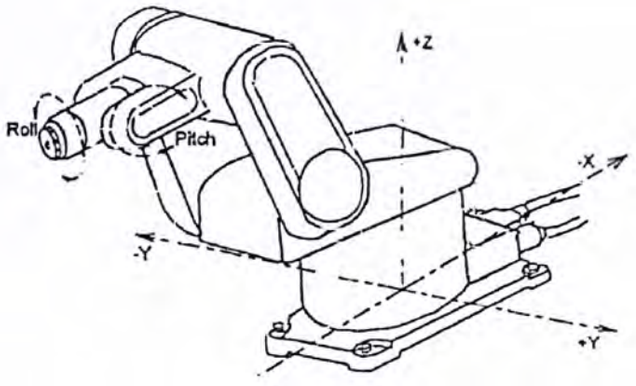


Figure 3-12 Cartesian coordinate system of the robot arm

The sensor board and one MICA Mote are mounted on the tool end of the robot. The body frame of the "mobile unit" is placed about the same as the Cartesian coordinate frame of the robot. And the sensing direction of the accelerometer is the same as y-axis of the Cartesian coordinate frame. The inertial coordinate frame is set the same as the initial position and orientation of body frame.

As all the points are represented in the two coordinate systems, it is unable to control the tool end to move an exact distance in terms of meter directly. Therefore the relationship between the real distance and the coordinate unit should be known first. Several calibrations are done to identify the ratio of the joint unit and Cartesian unit. The results are listed below.

Axes	Start unit	Finish unit	Difference	Real distance	Ratio
x-axis	406343	446344	40001	42 (mm)	0.00105 (mm/unit)
y-axis	100031	-100075	200106	257 (mm)	0.00128 (mm/unit)
z-axis	375211	225711	149500	149 (mm)	0.0001 (mm/unit)
r-axis rotation	53979	306021	360000	360 (degree)	0.001 (degree/unit)
axis 1	107252	137252	30000	43.5 (degree)	0.00145 (degree/unit)
axis 4	-140073	-16573	123500	180 (degree)	0.001457(degree/unit)

Table 3-2 Ratio of real distance and coordinate unit

3.4 Initial calibration and filtering

3.4.1 Convert ADC reading to acceleration

The sampled data by the 10-bit ADC of the MICA Mote are numbers which range from 0 to 1023. To obtain the relation between these numbers and the real acceleration data, initial calibration is needed.

The accelerometer can measure both static and dynamic accelerations. Besides, the output is proportional to the acceleration. By measuring well known static acceleration, gravity ($9.80708 m/s^2$), the scale factor related to the ADC data and the acceleration can be estimated. Two data are taken. One is putting the sensing direction of the accelerometer the same as the gravity. The other is completely opposite to the gravity. Then the scale factor (*SF*) is obtained as follow.

$$SF = \frac{2g}{a_{-1g} - a_{+1g}} \tag{3-6}$$

Where $g = 9.80708 m/s^2$, a_{+1g} is the ADC reading when pointing to the gravity and a_{-1g} is the ADC reading when opposite to the gravity. As both readings are the minimum and maximum readings respectively when the accelerometer is stationary, the tests are done several times to enhance the accuracy.

Theoretically, the reading when there is no acceleration applied to the accelerometer is

$$a_{0g} = \frac{a_{-1g} + a_{+1g}}{2} \tag{3-7}$$

However, due to the slight non-linearity of the accelerometer, the reading of zero acceleration may vary from the result in equation (3-7). To overcome this, the experimental data of zero acceleration is usually taken as a reference. Then the scale factors are adjusted from this point.

With this information, the readings from the ADC can be converted to accelerations by the following equation.

$$A = (reading - a_{0g}) \times SF \tag{3-8}$$

3.4.2 Identify configuration error

One of the error sources of the system is the configuration error. In the design of the system, the sensing direction of the accelerometer coincides with the y-axis of the body frame. However, misalignment exists. This leads inaccurate sensed acceleration. The figure below illustrates the misalignment.

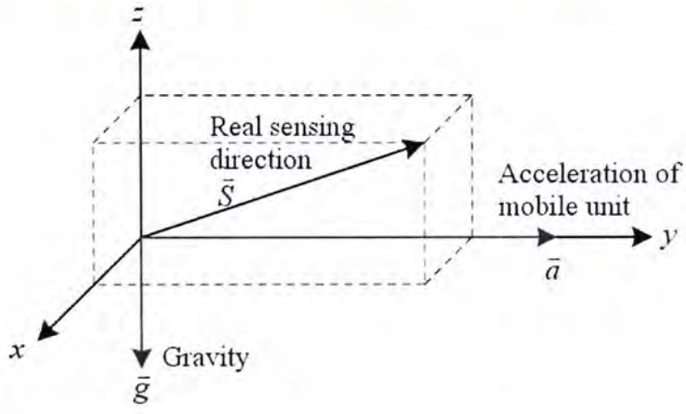


Figure 3-13 Misalignment of the sensing direction

Assume the sensing direction in unit vector form is $\bar{S} = [x_1, y_1, z_1]$, Then the sensed data is $\bar{S} \cdot \bar{a} + \bar{S} \cdot \bar{g}$. Here a method is proposed to compensate this error. Two calibration tests are performed before use. Firstly, the sensor board is arranged the same as the Figure 3-13 without any acceleration of the mobile unit. The component z_1 of $\bar{S}$ can be found as the value of gravity is known.

$$reading = \bar{S} \cdot \bar{g} = z_1 g$$

$$z_1 = \frac{reading}{g} \tag{3-9}$$

Then rotate the sensor board along x axis by 90 degree. By the same principle, the component y_1 of $\bar{S}$ can also be found.

$$reading = \bar{S} \cdot \bar{g} = y_1 g$$

$$y_1 = \frac{reading}{g} \tag{3-10}$$

As both components are known, this desired acceleration can be reconstructed. When the mobile unit is moving, the reading of the accelerometer consists of,

$$reading = \bar{S} \cdot \bar{a} + \bar{S} \cdot \bar{g} = y_1 a + z_1 g$$

The acceleration of the mobile unit is,

$$a = \frac{reading - z_1 g}{y_1} \tag{3-11}$$

3.4.3 Implement low pass filter

A low pass filter is utilized to remove the high frequency noise. A second order low pass Butterworth filter is designed. The transfer function in z domain is,

$$H(z) = \frac{b(1) + b(2)z^{-1} + b(3)z^{-2}}{1 + a(2)z^{-1} + a(3)z^{-2}}$$

The cutoff frequency is set at 10Hz. It is based on the observation in Figure 3-14. The cutoff frequency is the frequency where the magnitude response of the filter is $\sqrt{1/2}$. The design tool is MATLAB and the result is as follow.

$$H(z) = \frac{0.0675 + 0.1349z^{-1} + 0.0675z^{-2}}{1 - 1.143z^{-1} + 0.4128z^{-2}} \tag{3-12}$$

The sensed data is passed through the low pass filter first to reduce the effect of high frequency noise. The process is done in MATLAB. Then the data is

compensated for the configuration error which described in the previous section. The double integrations are performed afterward and the results are showed in the next section.

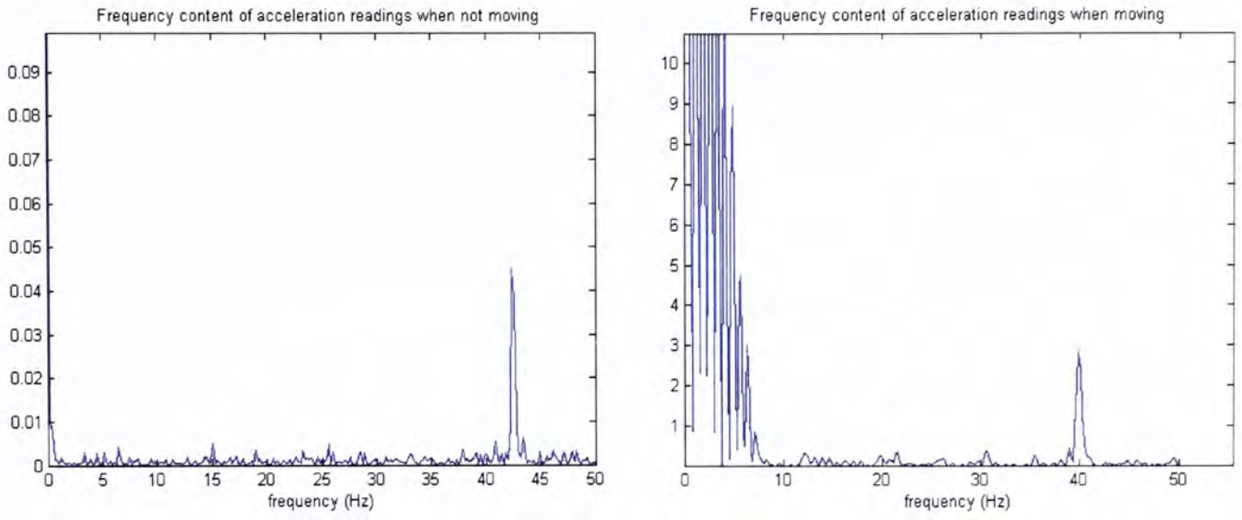


Figure 3-14 Frequency content of the acceleration readings in steady and moving state

3.5 Experimental results

3.5.1 Results

The senor board and a MICA Mote are mounted on the tool end of the robot arm. Another mote is connected to the master computer through interface board. Figure 3-15 illustrates the setup.

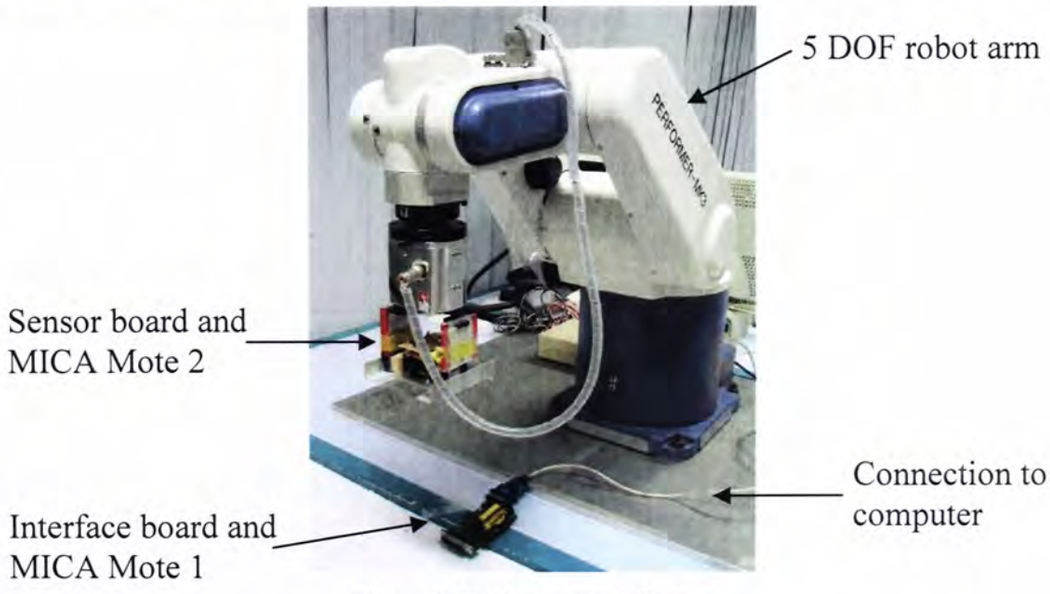


Figure 3-15 Experimental setup

The inertial frame is set the same as the Cartesian coordinate of the robot arm with center at the initial position of the tool end. The initial center and orientation of body frame is the same as the inertial frame. As this is one dimensional case, transformation between frames is not necessary. The sensing direction of the accelerometer is set at the same as y-axis of inertial frame. Initial calibrations described in section 3.4 are done and the results are listed in Table 3-3.

Term	a_{+1g}	a_{-1g}	a_{0g}	*SF for +ve a*	*SF for –ve a*	z_1	y_1
Value	314	707	517	0.1032324	0.0966215	-0.0980708	0.99507703

Table 3-3 Results of initial calibrations

The robot moves a return trip along y-axis with distance about 0.257m. The sensed acceleration data is shown below.

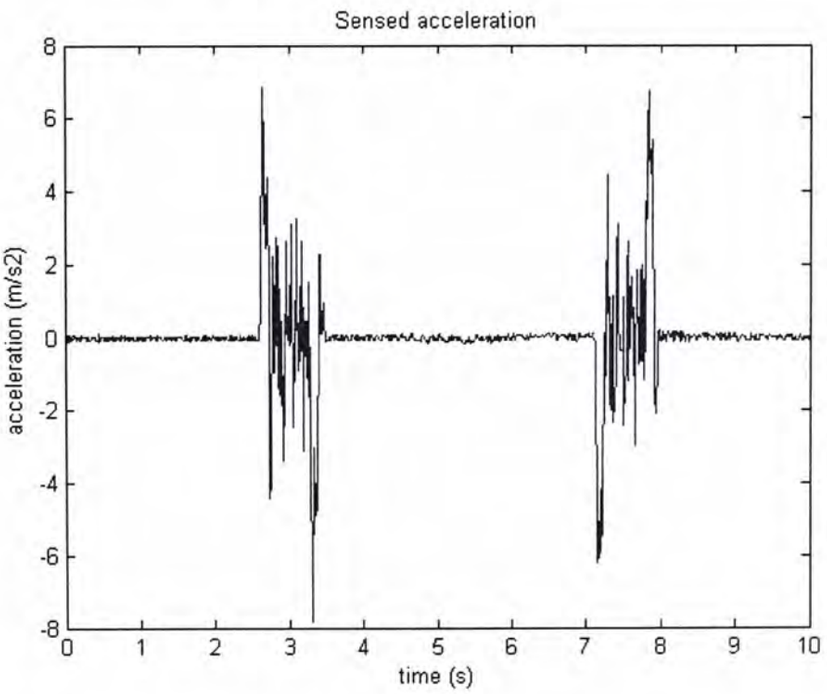


Figure 3-16 Sensed acceleration

The compensation of configuration error and low pass filter are implemented. Double numeric integrations are applied to find the velocity and distance traveled. The results are shown in the following figures.

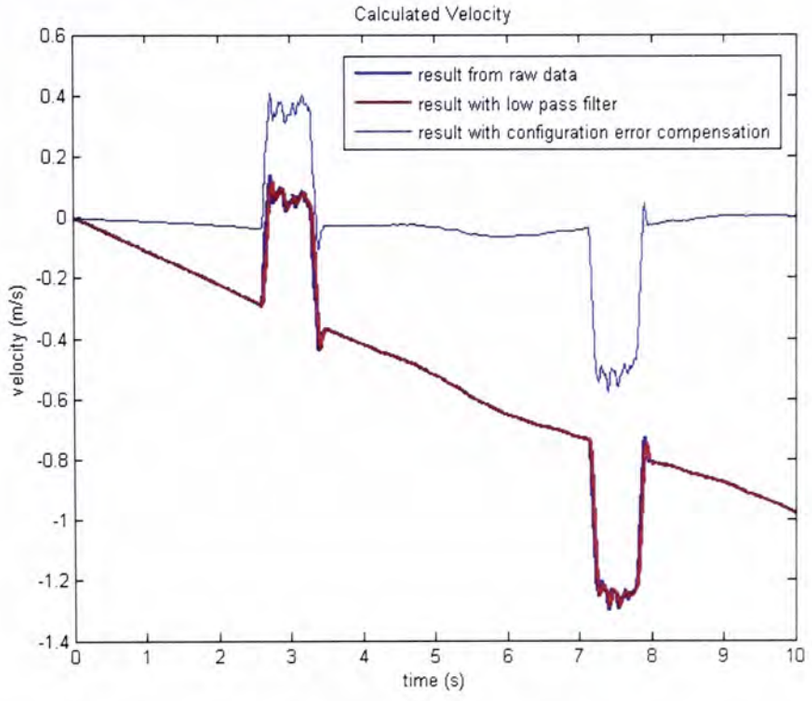


Figure 3-17 Calculated velocity

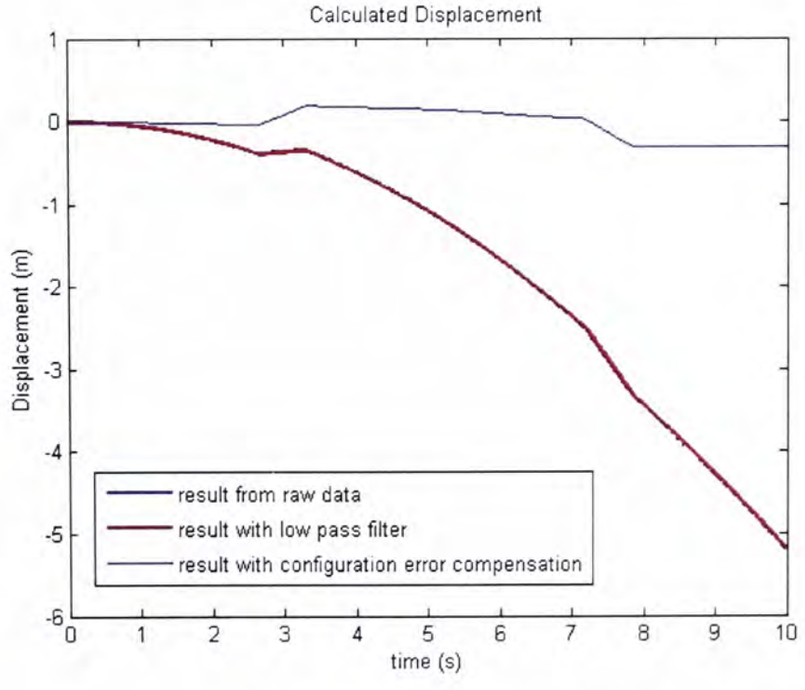


Figure 3-18 Calculated displacement

Some values of the calculated position are listed in the following table.

	Real position	**From raw data**	**With low pass filter**	**With configuration error compensation**
After 2s	0	-0.2187	-0.2157	-0.0247
	Error	**0.2187**	**0.2157**	**0.0247**
After 5s	0.257	-1.0766	-1.0697	0.1451
	Error	**1.3336**	**1.3267**	**0.1119**
After 10s	0	-5.2028	-5.1906	-0.3106
	Error	**5.2028**	**5.1906**	**0.3106**

Table 3-4 Values of calculated position (in meter)

3.5.2 Discussion

From the experimental results listed in Table 3-4, the error of displacement which is directly obtained from raw data is very large. After 10 seconds, the error is more than 5 meters. This indicates that the raw acceleration not only contains the acceleration we concern. In other words, directly double integrating from the raw acceleration is not an effective way to obtain the displacement. The data should be undergone some forms of signal processing to reduce the noise and extract the meaningful acceleration.

The implementation of low pass filter can reduce the high frequency noise. The results show that the performance is slightly improved by the filter. However, the error is still large that the result is unacceptable. The performance of the low pass filter in accuracy enhancement is not effective. This may be due to the less significant magnitude of the high frequency noise compared to the actual acceleration. On the other hand, the noise with lower frequency can not be filtered in this case. In the next chapters, the low pass filter is replaced by Kalman filter.

The configuration error of the setup is one of the major error sources from this experiment. Since a small error in acceleration leads large position error in a ratio of t^2, the performance of the system is greatly improved when the configuration error is compensated. However, the results are still not accurate enough. One of the reason is the slight vibration of the acceleration signal even the mobile unit is not moved. Figure 3-19 shows the effect of the slight vibration of acceleration data on the calculated result. It is a kind of measuring error occurs in data sampling process.

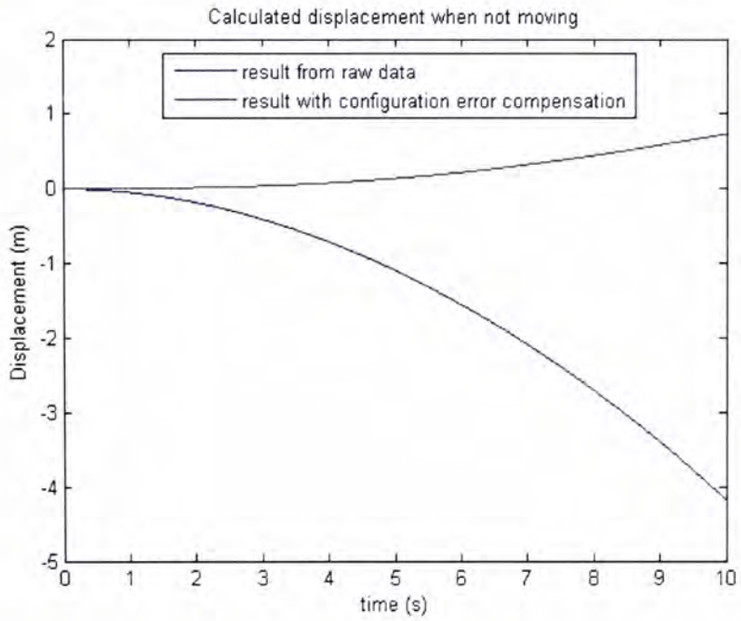


Figure 3-19 Calculated displacement due to the slight vibration of acceleration data

Another reason is the inaccurate in the accelerometer itself. The output reading can be affected by several items and is modeled as following [39],

$$\tilde{a} = (1+s)a + b + n \tag{3-13}$$

Where $\tilde{a}$ is the output reading, a is the actual acceleration, s is the scale factor error, b is the zero bias error and n is the random noise. To achieve an accuracy system, these error sources should be identified and compensated.

To conclude, this experiment shows that the output of the MEMS accelerometer is not accurate enough for direct usage in position determination. Different signal

processing methods should be implemented to enhance the accuracy. This is the next step of this project.

3.6 Summary

To evaluate the performance of MEMS accelerometer in position determination, a one dimensional INS is constructed with one MEMS accelerometer. The detail and experimental results of the INS are shown in this chapter.

The MEMS accelerometer utilized is ADXL103 from Analog Devices Inc.. The properties are described. Based on the specifications, a circuit is designed. On the other hand, integrated device MICA Mote is utilized. It contains a microprocessor, eight channels ADC, 512 kbyte flash memory and a radio module. A mote is connected to the sensor board while another is connected to the master computer. Both motes can communicate wirelessly. These form a one-dimensional INS.

Initial calibration of the accelerometer utilizing gravity is introduced. The method of configuration error compensation and low pass filtering are proposed to enhance the accuracy.

The setup is tested in robot platform which can move in five DOF. The result of directly double integrated from raw data is the worst one. This indicates that signal processing is needed if accuracy is important. Low pass filtering and configuration error compensation are also performed. The latter is found that can reduce the error in more than ten times after ten seconds. However, the accuracy is still not good enough for position determination. Therefore, further methods to reduce the unwanted data are needed to investigate.

Chapter 4
Performance Improvement

The performance of the accelerometer in one dimensional INS was shown in the last chapter. The result indicates that error reduction methodologies are needed to enhance the accuracy in position determination. In this chapter, two error reduction methods are discussed and implemented to improve the performance of the accelerometer.

4.1 Fuzzy logic based steady state detector

4.1.1 Principle

The velocity bias is the main source of the error. It is due to small inaccurate acceleration data. As the position is got from integration of the velocity signal, even the mobile unit is not moving, a small bias in velocity results in "moving" in determined position.

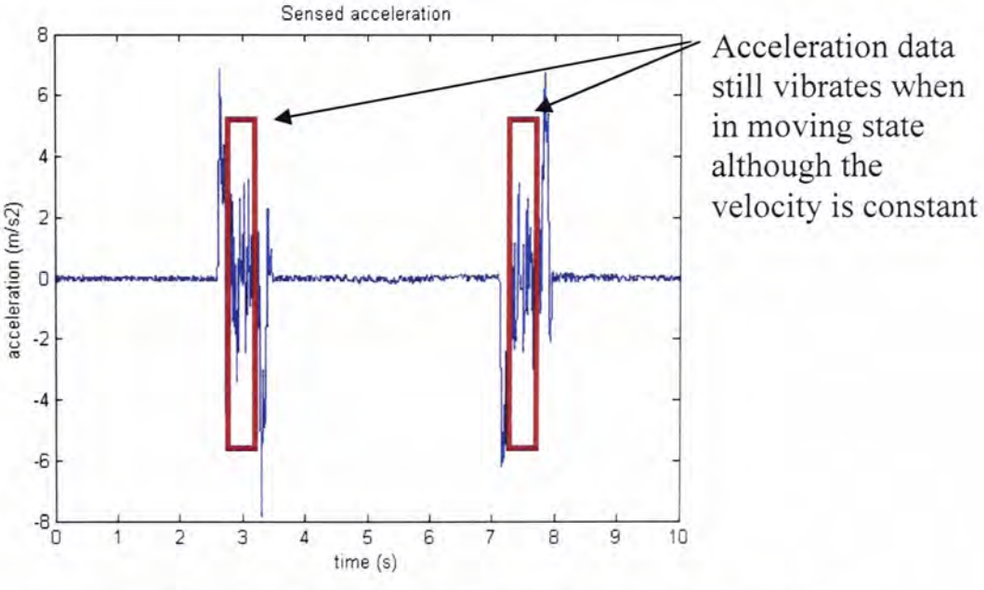


Figure 4-1 Illustration of vibration of acceleration when the mobile unit is moving

This problem can be reduced if some constraints of the system are known. In this system, the vibration of the acceleration signal is lower in steady state than that in moving state. Although the mobile unit moves at constant velocity, the vibration is still large, as shown in Figure 4-1. So vibration is one characteristic that indicates the motion state of the mobile unit. If the state of not moving is known, the acceleration and velocity signals can be calibrated to be zero. Then the effect of the velocity shift on the position detected can be reduced.

Therefore, a steady state detector based on fuzzy logic is proposed. The aim of the detector is to determine when the mobile unit is not moving. The standard deviation of the last 16 readings is utilized to evaluate the state of the mobile unit. The definition of standard deviation is as follow,

$$\sigma = \sqrt{\frac{\sum_{i=1}^{16}(a(i)-\mu)^2}{16}} \tag{4-1}$$

Where σ is the standard deviation, $a(i)$ is the acceleration reading and μ is the mean of last 16 readings. Figure 4-2 below shows the significant difference in standard deviations between moving and steady state.

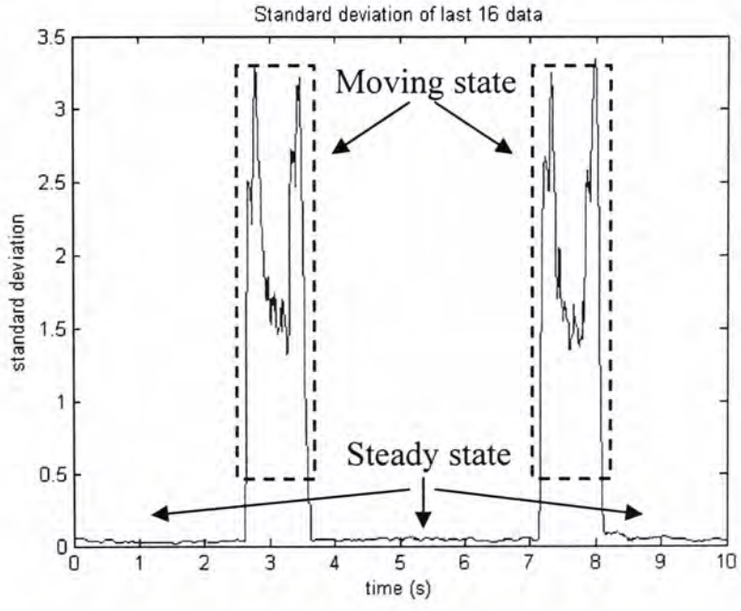


Figure 4-2 Standard deviation of last 16 data

Based on the values of standard deviation of last 16 acceleration readings, two fuzzy rules are designed as following,

If standard deviation is small, then the mobile unit is not moving;

If standard deviation is large, then the mobile unit is moving

The designed membership function of the input is shown below.

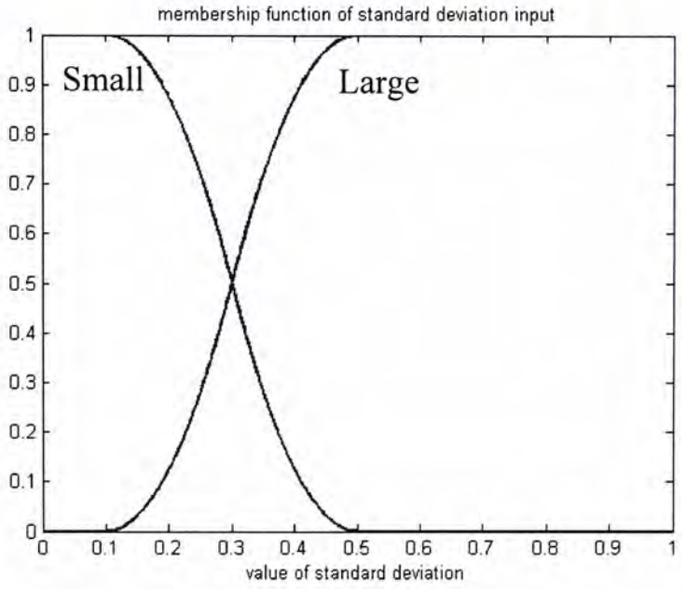


Figure 4-3 Membership function of the input

If the mobile unit is not moving, the determined velocity is calibrated to be zero.

4.1.2 Experimental result

The acceleration data obtained in section 3.5 is utilized here to investigate the performance of the fuzzy logic based steady state detector. The raw data is firstly compensated for the configuration error. Then the data is passed through the detector to verify the state of the mobile unit. If it is in steady state, both acceleration and velocity are set to zero. The comparison of the performance between with and without the detector is shown in Figure 4-4 and Table 4-1.

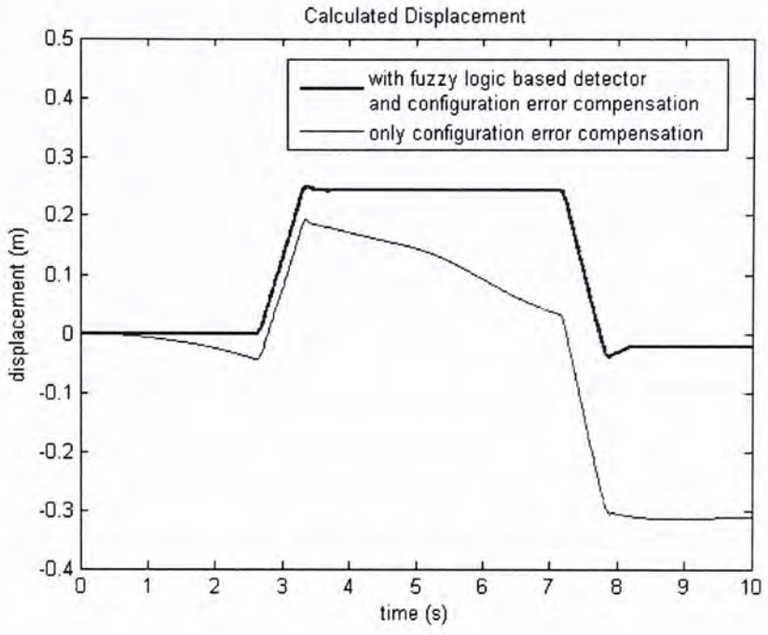


Figure 4-4 Performance of fuzzy logic based state detector

	Real position	**Only configuration error compensation**	**With fuzzy logic based detector and configuration error compensation**
After 2s	0	-0.0247	0
	Error	**0.0247**	**0**
After 5s	0.257	0.1451	0.2429
	Error	**0.1119**	**0.0141**
After 10s	0	-0.3106	-0.0222
	Error	**0.3106**	**0.0222**

Table 4-1 Comparison of the performance between with and without the detector

The results show that the error is greatly reduced by ten times after ten seconds when the fuzzy logic based detector and the configuration error compensation are implemented together. Actually, the detector identifies the slight vibration of acceleration successfully. The performance is acceptable in position determination in a short period of time.

4.2 Kalman filtering

As stated in equation (3.13), the output of accelerometer contains random noise. According to [51], the noise of ADXL103 has the characteristics of white Gaussian noise, which contributes equally at all frequencies. The noise with similar frequency as the motion can not be reduced by the low pass filter described in section 3.4.3. To solve this, a discrete Kalman filter is utilized. Kalman filter is a common method for random noise reduction and data fusion. It combines the knowledge of the statistical nature of system noise and system dynamics to estimate the state of the system. Some similar works can be found in [41].

4.2.1 Discrete Kalman filter

Assume the state to be estimated can be modeled in the form

$$x_{k+1} = \Phi_k x_k + w_k \tag{4-2}$$

And the measurement of the process is assumed to occur at discrete points in time in accordance with the linear relationship

$$z_k = H_k x_k + v_k \tag{4-3}$$

Where

k indicates the value at time k,

x is state vector,

Φ is matrix relating x_k to x_{k+1} in the absence of a forcing function,

w is assumed to be a white sequence with covariance Q,

z is the measurement,

H is matrix giving the ideal (noiseless) connection between the measurement and the state,

v is assumed to be a white sequence measurement error with covariance R.

The objective of the Kalman filter is to estimate the state of the system x. The estimated state is represented by $\hat{x}$. The process of the estimation is presented here and the deviation of equations can be found in [52].

Kalman filtering is a two step process, prediction and correction. The correction step makes corrections to an estimate, based on new information from the sensor measurement. The estimated state is corrected by this equation,

$$\hat{x}_k(+) = \hat{x}_k(-) + K_k(z_k - H_k\hat{x}_k(-)) \tag{4-4}$$

Where (+) indicates the corrected value while (-) is the predicted value. K_k is the Kalman gain which is defined as,

$$K_k = P_k(-)H_k^T(H_kP_k(-)H_k^T + R_k)^{-1} \tag{4-5}$$

Where P_k is the covariance matrix. The correction of P_k is,

$$P_k(+) = P_k(-) - K_kH_kP_k(-) \tag{4-6}$$

In prediction step, both state and its associated covariance matrix are predicted according to the corrected state estimate and covariance matrix from one time step before. The equation to predict the state is,

$$\hat{x}_{k+1}(-) = \Phi_k x_k(+) \tag{4-7}$$

While the prediction of covariance matrix is,

$$P_{k+1}(-) = \Phi_k P_k(+)\Phi_k^T + Q_k \tag{4-8}$$

The process can be started in either prediction stage or correction stage. When starting from correction stage, $P_0(-)$ and $\hat{x}_0(-)$ are initialized. Then the process of the filtering at time k is shown in Figure 4-5.

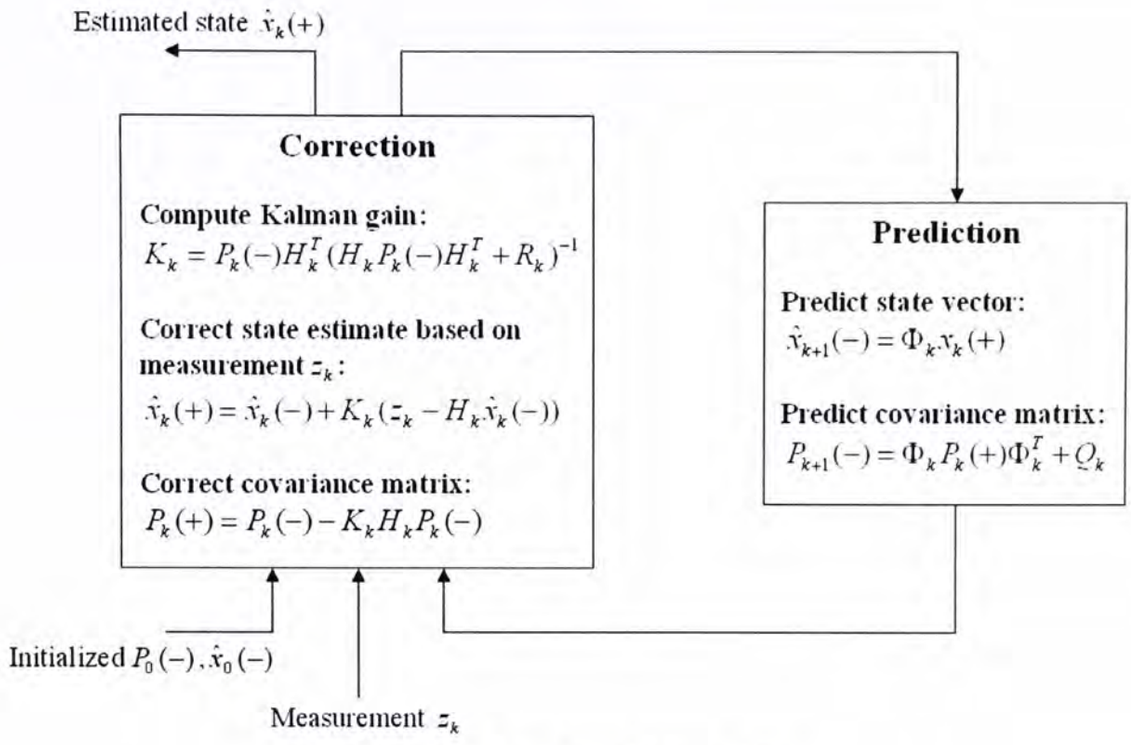


Figure 4-5 Data flow of Kalman filter

4.2.2 Combine with fuzzy logic based steady state detector

One purpose of Kalman filtering is to estimate the actual signal from a noisy signal. With modeled process noise covariance Q and measurement noise covariance R, the effect of noise can be reduced. On the other hand, the fuzzy logic based state detector can reduce the small variation when the mobile unit is stationary. It is a nice complement if the state detector is combined with the Kalman filter. Then the noise can be reduced in both steady and moving state.

Some works of Kalman filtering on acceleration data under white Gaussian noise can be found in [41]. At that case, acceleration is the only measurement data that inputted to the filter. This method is modified here that another measurement data is added to enhance the performance of the estimation.

The resulting velocity from the fuzzy logic based steady state detector is the additional measurement data. The velocity of the mobile unit is zero if it is in

steady state. This information is then inputted to the Kalman filter as a reference to improve the estimation of the position. Therefore, the measurement is,

$$z_k = \begin{bmatrix} v_{fk} \\ a_{fk} \end{bmatrix} \tag{4-9}$$

Where v_{fk} and a_{fk} are the velocity and acceleration output of the steady state detector respectively. The state of the system can be represented by,

$$x_k = [d_k \quad v_k \quad a_k]^T \tag{4-10}$$

The measurement sensitivity matrix H_k is,

$$H_k = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4-11}$$

With sampling time Δt, the state transition matrix Φ_k is,

$$\Phi_k = \begin{bmatrix} 1 & \Delta t & \Delta t^2/2 \\ 0 & 1 & \Delta t \\ 0 & 0 & 1 \end{bmatrix} \tag{4-12}$$

The process noise covariance of the white Gaussian noise in this model is the same as that in [41],

$$Q_k = \begin{bmatrix} \frac{W}{20}\Delta t^5 & \frac{W}{8}\Delta t^4 & \frac{W}{6}\Delta t^3 \\ \frac{W}{8}\Delta t^4 & \frac{W}{3}\Delta t^3 & \frac{W}{2}\Delta t^2 \\ \frac{W}{6}\Delta t^3 & \frac{W}{2}\Delta t^2 & W\Delta t \end{bmatrix} \tag{4-13}$$

Where W is the power spectral density of the input white noise.

The measurement noise covariance R_k is defined as,

$$R_k = \begin{bmatrix} R_v & 0 \\ 0 & R_a \end{bmatrix} \tag{4-14}$$

R_v and R_a are the noise of velocity and acceleration signals. They are related to the noise of accelerometer which can be found in data sheet or experiment.

4.2.3 Experimental results

I. Experiment 1

To illustrate the performance of the error reduction methods, experiment with longer traveled distance than that in section 3.5 is performed. The robot arm travels a return trip with distance 0.388m. The sensed data is firstly compensated the effect of configuration error. Then the data is passed through the steady state detector. The detector corrects the velocity and acceleration when it is not moving. Both data are the measurement inputs of the Kalman filter.

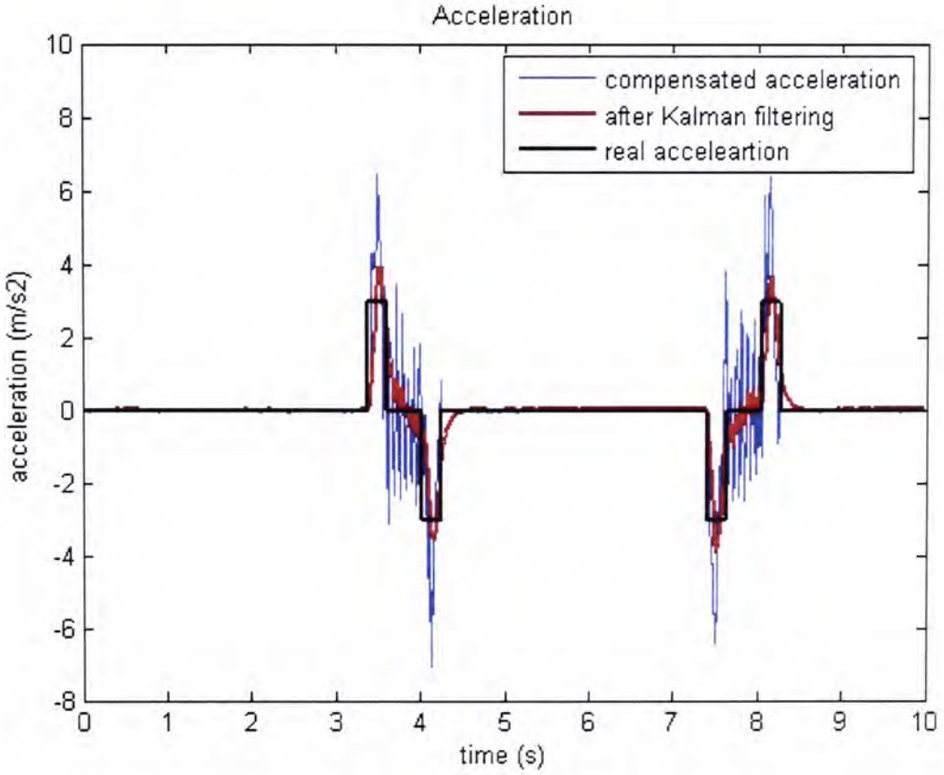


Figure 4-6 Sensed acceleration

The resulting acceleration data, as shown in Figure 4-6, is in clearer shape than the original one. This is because the effect of random white Gaussian noise is reduced. The calculated displacement of the motion is shown in the following figure.

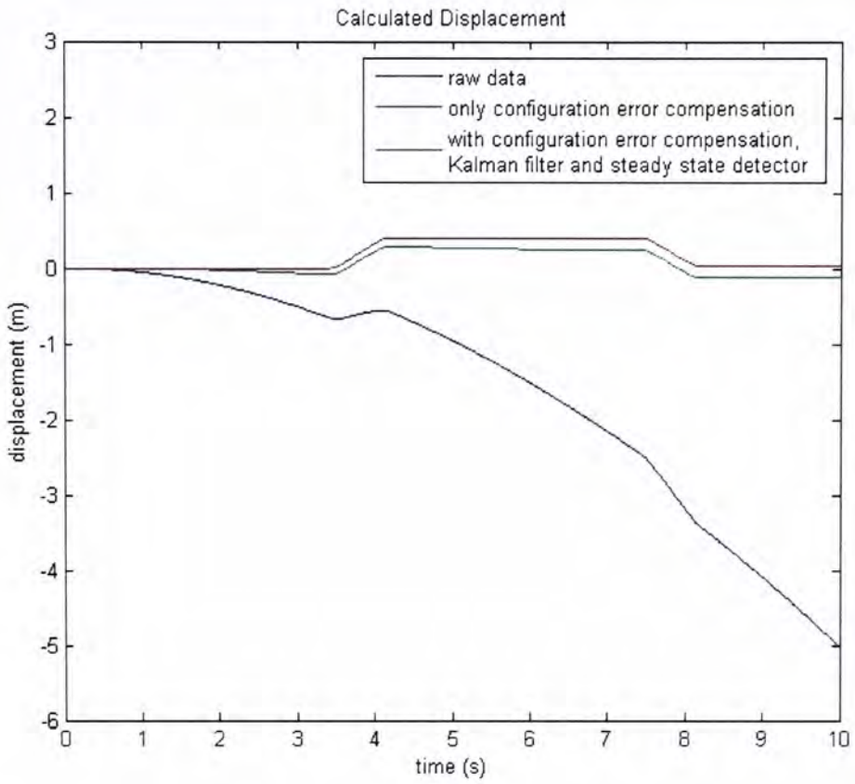


Figure 4-7 Calculated displacement with different error reduction methods

The details of the calculated results are shown in Table 4-2.

	Real position	**Only configuration error compensation**	**With configuration error compensation, Kalman filter and steady state detector**
After 2s	0	-0.0254	0
	Error	**0.0254**	**0**
After 5s	0.388	0.2717	0.4035
	Error	**0.1163**	**0.0155**
After 10s	0	-0.1094	0.0346
	Error	**0.1094**	**0.0346**

Table 4-2 Comparison of the performance of different error reduction methods

The error of the calculated displacement from raw data is about 5 meters after 10 seconds. The result is improved by compensating the configuration error. The error is further reduced when the Kalman filter and steady state detector are utilized on the compensated data. This experiment shows the performance of the error reduction methods proposed in this chapter.

II. Experiment 2

In the last experiment, both motions are in the same velocities and accelerations. Here, the setup is tested with two motions with different velocities. The robot moves a return trip with distance 0.257m. The maximum velocity of the first motion is set at $0.3\ m/s$ while the maximum velocity of second motion is $0.2\ m/s$. The sensed acceleration is shown in Figure 4-8.

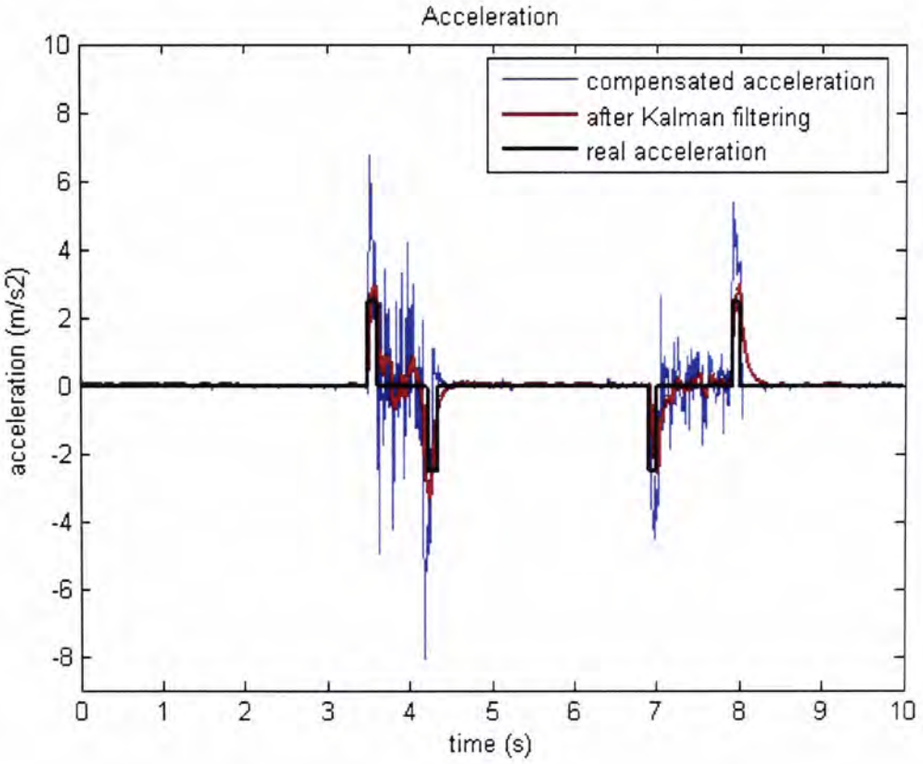


Figure 4-8 Sensed acceleration

The calculated displacement is shown in Figure 4-9 and the details are listed in Table 4-2.

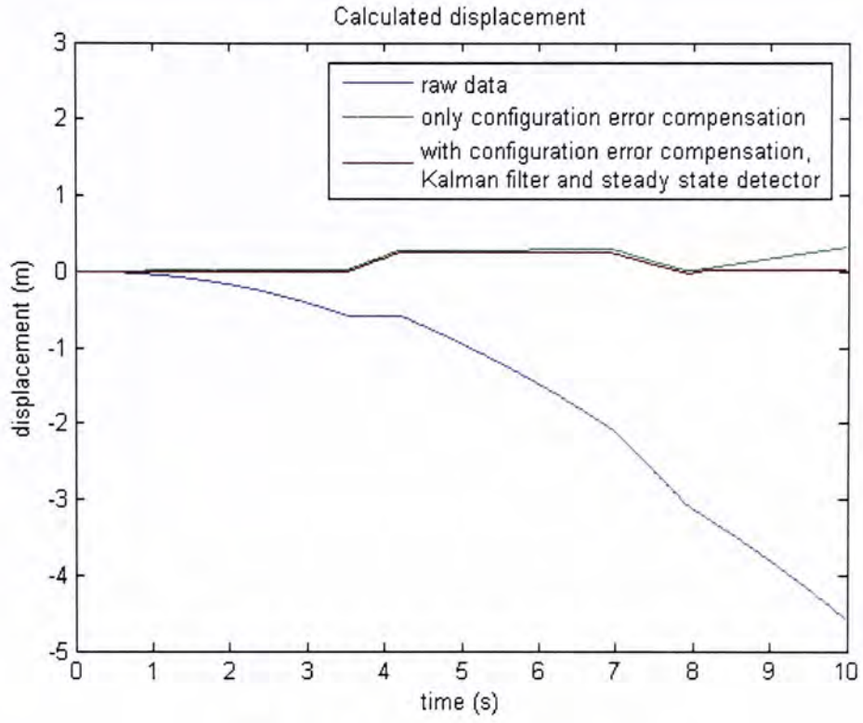


Figure 4-9 Calculated displacement

	Real position	**Only configuration error compensation**	**With configuration error compensation, Kalman filter and steady state detector**
After 2s	0	0.0207	0
	Error	**0.0207**	**0**
After 5s	0.257	0.2787	0.2419
	Error	**0.0217**	**0.0151**
After 10s	0	0.3171	0.0196
	Error	**0.3171**	**0.0196**

Figure 4-10 Comparison of the performance of different error reduction methods

The result is highly improved when the error reduction methods presented in this chapter are utilized. These two experiments show the methods are workable to enhance the accuracy of the calculated displacement. On the other hand, we can conclude that with suitable signal processing, the performance of MEMS accelerometer is acceptable as a distance measurement device in a short duration.

4.3 Summary

In the previous chapter, the performance of the accelerometer is not satisfied in position determination. Therefore, two noise reduction methods are proposed in this chapter to improve the performance of the accelerometer.

The first one is the steady state detector based on fuzzy logic. The aim of the detector is to identify whether the mobile unit is moving or stationary. Its principle is based on the knowledge that the vibration of acceleration data is much larger in moving state than that in steady state. This property holds even the velocity is constant and the acceleration is zero. The velocity and acceleration data are set to be zero if the unit is detected stationary. The experimental results show that the detector can remove the small noise when the unit is steady and the performance is greatly improved.

Another method is utilizing Kalman filter to reduce the random white Gaussian noise of the accelerometer. The aim of the Kalman filter is to remove the noise when the unit is moving. It combines with the state detector motioned above to give a nice noise filtering on the raw acceleration data.

The results in this chapter show that the MEMS accelerometer can be an acceptable distance measurement device in a short period of time. The MEMS accelerometer is then utilized to construct a GF-INS. The noise reduction methods presented in this chapter are implemented in the system GF-INS. The details are shown in the next two chapters.

Chapter 5
Construction of GF-INS

The performance of the MEMS accelerometer in one dimensional INS is shown in previous chapters. The work is extended to construct a GF-INS based on MEMS accelerometers. A brief introduction to the development of utilizing only linear accelerometers for motion detection is given in section 2.4. The algorithm proposed by Tan and Park [45] is studied. This six accelerometers configuration can replace traditional three-accelerometers-three-gyroscopes INS.

5.1 Principle of GF-INS

5.1.1 Algorithm

The algorithm presented in this section is originally developed by Tan and Park. The detail can be found in [45].

The derivation starts from analyzing the output of accelerometer. The output of the accelerometer can be modeled as,

$$A = \left\langle \ddot{r}_I - a_g + \delta\ddot{\theta}_I, \theta_I \right\rangle \tag{5-1}$$

It consists of three parts,

$\left\langle \ddot{r}_I, \theta_I \right\rangle$ is the projection of acceleration of the center of mass onto the sensing direction,

$\left\langle a_g, \theta_I \right\rangle$ is the projection of gravitational acceleration onto the sensing direction,

$\left\langle \delta\ddot{\theta}_I, \theta_I \right\rangle$ is the projection of the product of the displacement and change in the sensing direction over time onto the sensing direction.

Since the typical value of δ is about 10^{-7} m, the output of the accelerometer can be simplified as,

$$A = \left\langle \ddot{r}_I - a_g, \theta_I \right\rangle \tag{5-1}$$

Then consider the motion of rigid body. Assume an accelerometer is mounted on point M on the rigid body. The bases of the inertial frame and the body frame are $\{x_1, y_1, z_1\}$ and $\{x_2, y_2, z_2\}$ respectively, as shown in Figure 5-1.

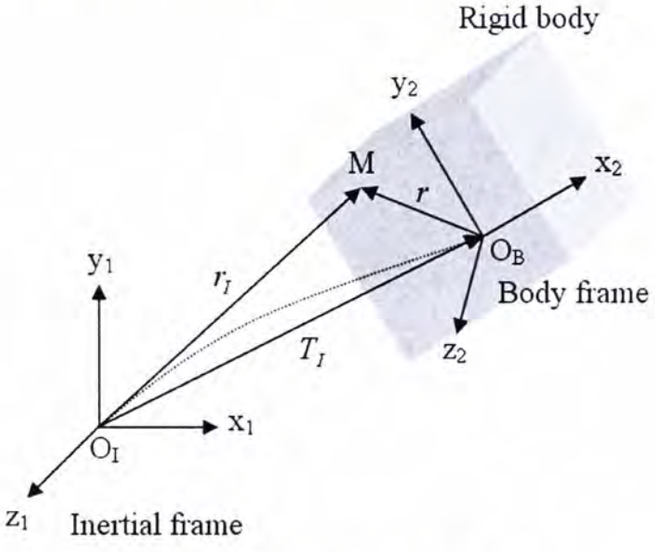


Figure 5-1 Motion of rigid body

From the graph,

$$r_I = T_I + r \tag{5-2}$$

Where T_I is the translation matrix. All these three components are respect to inertial frame. Assume there is rotation matrix R such that,

$$\begin{bmatrix} x_1 & y_1 & z_1 \end{bmatrix}^T = R \begin{bmatrix} x_2 & y_2 & z_2 \end{bmatrix}^T \tag{5-3}$$

Let the coordinates of point M with respect to body frame be u, then

$$r = Ru \tag{5-4}$$

Put it back to equation (5-2),

$$r_I = T_I + Ru \tag{5-5}$$

Therefore, the acceleration of point M with respect to the inertial frame is

$$\ddot{r}_I = \ddot{T}_I + \ddot{R}u \tag{5-6}$$

Then by the properties of rotation matrix,

$$R^T R = I$$
$$R^T \dot{R} + \dot{R}^T R = 0 \tag{5-7}$$

Define a skew-symmetric matrix Ω such that $\Omega = R^T \dot{R}$ and $\Omega^T = -\Omega$, the equation (5-7) can be simplified as,

$$\dot{R} = R\Omega$$
$$\ddot{R} = R\dot{\Omega} + \dot{R}\Omega$$
$$\ddot{R} = R\left(\dot{\Omega} + \Omega^2\right) \tag{5-8}$$

The angular velocity $\omega = \begin{bmatrix}\omega_x & \omega_y & \omega_z\end{bmatrix}^T$ is the cross product equivalence of Ω, such that

$$\Omega = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \tag{5-9}$$

Then combine equations (5-6) and (5-8), the acceleration of the point M is

$$\ddot{r}_I = \ddot{T}_I + R\left(\dot{\Omega} + \Omega^2\right)u \tag{5-10}$$

The sensed acceleration from the accelerometer can be modeled by combining equations (5-1), (5-10) and $\theta_I = R\theta$,

$$\begin{aligned} A(u,\theta) &= \left\langle \ddot{T}_I + R\left(\dot{\Omega} + \Omega^2\right)u - a_g, R\theta \right\rangle \\ &= \left\langle \left(\ddot{T}_I - a_g\right) + R\left(\dot{\Omega} + \Omega^2\right)u, R\theta \right\rangle \\ &= \left\langle R^T\left(\ddot{T}_I - a_g\right) + \left(\dot{\Omega} + \Omega^2\right)u, \theta \right\rangle \\ &= \left\langle P, \theta \right\rangle + \left\langle Gu, \theta \right\rangle \end{aligned} \tag{5-11}$$

Where $P = R^T\left(\ddot{T}_I - a_g\right)$ and $G = \dot{\Omega} + \Omega^2$. $\langle P, \theta \rangle$ is the linear acceleration which is projected to sensing direction of the accelerometer. While $\langle Gu, \theta \rangle$ is the angular acceleration, which contains tangential (skew-symmetric $\dot{\Omega}$) and centripetal (symmetric Ω^2) accelerations.

Assume there are N accelerometers on the rigid body. The position and sensing direction of the accelerometers are represented by u and θ respectively. Then define three variables,

$$J_1 = \begin{bmatrix} u_1 \times \theta_1 & \cdots & u_N \times \theta_N \end{bmatrix} \tag{5-12}$$

$$J_2 = \begin{bmatrix} \theta_1 & \cdots & \theta_N \end{bmatrix} \tag{5-13}$$

$$J = \begin{bmatrix} J_1^T & J_2^T \end{bmatrix} \tag{5-14}$$

From (5-11)

$$\begin{aligned} A_i(u_i, \theta_i) &= \left\langle R^T\left(\ddot{T}_I - a_g\right) + \left(\dot{\Omega} + \Omega^2\right)u_i, \theta_i \right\rangle \\ &= \left\langle P + \dot{\omega} \times u_i + \Omega^2 u_i, \theta_i \right\rangle \\ &= \left\langle P, \theta_i \right\rangle + \left\langle \dot{\omega}, u_i \times \theta_i \right\rangle + \left\langle \Omega^2 u_i, \theta_i \right\rangle \\ &= \theta_i^T P + \left(u_i \times \theta_i\right)^T \dot{\omega} + \theta_i^T \Omega^2 u_i \\ &= \begin{bmatrix} \left(u_i \times \theta_i\right)^T & \theta_i^T \end{bmatrix} \begin{bmatrix} \dot{\omega} \\ P \end{bmatrix} + \begin{bmatrix} \theta_i^T \Omega^2 u_i \end{bmatrix} \end{aligned} \tag{5-15}$$

So

$$A_i = \begin{bmatrix} A_1 \\ \vdots \\ A_N \end{bmatrix} = J \begin{bmatrix} \dot{\omega} \\ P \end{bmatrix} + \begin{bmatrix} \theta_1^T \Omega^2 u_1 \\ \vdots \\ \theta_N^T \Omega^2 u_N \end{bmatrix} \tag{5-16}$$

If J has a left inverse $Q \in R^{6\times N}$, then

$$\begin{bmatrix} \dot{\omega} \\ P \end{bmatrix} = Q \begin{bmatrix} A_1 \\ \vdots \\ A_N \end{bmatrix} - Q \begin{bmatrix} \theta_1^T \Omega^2 u_1 \\ \vdots \\ \theta_N^T \Omega^2 u_N \end{bmatrix} \tag{5-17}$$

Equation (5-17) shows that the angular accelerations and the linear accelerations can be related to a set of acceleration data. The problem is that it is still complicated to solve the angular and linear accelerations if only a set of accelerometers' readings is known. The contribution of Tan and Park is to figure out a special six accelerometers configuration in which the equation (5-17) can be simplified to a linear combination of the six accelerations. The particular configuration of the six accelerometers is as follow,

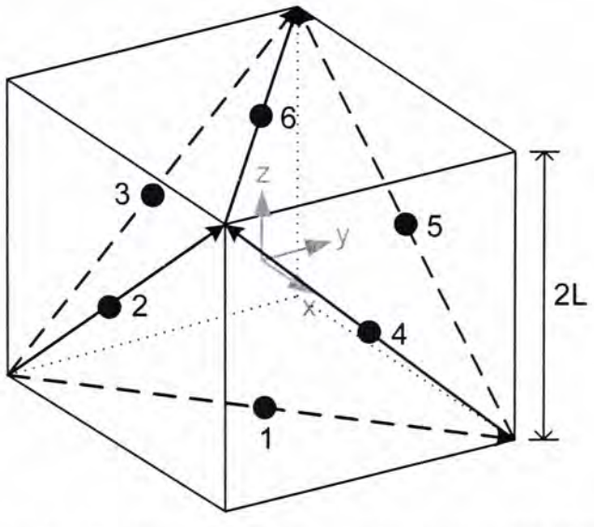


Figure 5-2 Configuration of six accelerometers GF-INS

With a cube shape, each accelerometer is placed at the center of the diagonal in each side. The center of the body frame is located at the center of the cube. The position and sensing direction of the accelerometers are represented as follow,

$$U = [u_1 \dots u_6] = L\begin{bmatrix} 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{5-18}$$

$$J_2 = [\theta_1 \dots \theta_6] = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 & 0 & 0 & -1 & -1 \\ 1 & 0 & 1 & -1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 \end{bmatrix} \tag{5-19}$$

$$J_1 = [u_1 \times \theta_1 \dots u_6 \times \theta_6] = \frac{L}{\sqrt{2}}\begin{bmatrix} 1 & -1 & 0 & 0 & 1 & -1 \\ -1 & 0 & 1 & -1 & 0 & -1 \\ 0 & 1 & -1 & -1 & 1 & 0 \end{bmatrix} \tag{5-20}$$

Where U is the position of the accelerometers and J_2 represents the sensing directions. With this configuration, the inverse of $J = \begin{bmatrix} J_1^T & J_2^T \end{bmatrix}$ is,

$$Q = J^{-1} = \frac{1}{2}\begin{bmatrix} \frac{1}{L^2} J_1 \\ J_2 \end{bmatrix} \tag{5-21}$$

Therefore, from equation (5-17),

$$\begin{bmatrix} \dot{\omega} \\ P \end{bmatrix} = Q\begin{bmatrix} A_1 \\ \vdots \\ A_6 \end{bmatrix} - Q\begin{bmatrix} \theta_1^T \Omega^2 u_1 \\ \vdots \\ \theta_6^T \Omega^2 u_6 \end{bmatrix}$$

$$= \frac{1}{2}\begin{bmatrix} \frac{1}{L^2} J_1 \\ J_2 \end{bmatrix}\begin{bmatrix} A_1 \\ \vdots \\ A_6 \end{bmatrix} - \frac{1}{2}\begin{bmatrix} \frac{1}{L^2} J_1 \\ J_2 \end{bmatrix}\left(- LJ_2{}^T \begin{bmatrix} \omega_2\omega_3 \\ \omega_1\omega_3 \\ \omega_1\omega_2 \end{bmatrix}\right)$$

$$= \frac{1}{2}\begin{bmatrix} \frac{1}{L^2} J_1 \\ J_2 \end{bmatrix}\begin{bmatrix} A_1 \\ \vdots \\ A_6 \end{bmatrix} - \frac{1}{2}\begin{bmatrix} \frac{-1}{L} J_1 J_2{}^T \\ -LJ_2 J_2{}^T \end{bmatrix}\begin{bmatrix} \omega_2\omega_3 \\ \omega_1\omega_3 \\ \omega_1\omega_2 \end{bmatrix}$$

$$= \frac{1}{2}\begin{bmatrix} \frac{1}{L^2} J_1 \\ J_2 \end{bmatrix}\begin{bmatrix} A_1 \\ \vdots \\ A_6 \end{bmatrix} - \begin{bmatrix} O_3 \\ LI_3 \end{bmatrix}\begin{bmatrix} \omega_2\omega_3 \\ \omega_1\omega_3 \\ \omega_1\omega_2 \end{bmatrix} \tag{5-22}$$

Put J_1 and J_2 into the above equation and simplify,

$$\begin{bmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\omega}_3 \end{bmatrix} = \frac{1}{2\sqrt{2}L} \begin{bmatrix} A_1 - A_2 + A_5 - A_6 \\ -A_1 + A_3 - A_4 - A_6 \\ A_2 - A_3 - A_4 + A_5 \end{bmatrix} \tag{5-23}$$

$$P = \frac{1}{2\sqrt{2}} \begin{bmatrix} A_1 + A_2 - A_5 - A_6 \\ A_1 + A_3 - A_4 + A_6 \\ A_2 + A_3 + A_4 + A_5 \end{bmatrix} + L \begin{bmatrix} \omega_2\omega_3 \\ \omega_1\omega_3 \\ \omega_1\omega_2 \end{bmatrix} \tag{5-24}$$

This system of equations illustrates the simple relation between the angular and linear motion of the rigid body and the readings of the six accelerometers. To obtain the linear and angular motion from the six accelerations, follow the procedures listed below.

1. Utilizing initial conditions $R(t_0)$ and $\dot{R}(t_0)$, obtain $\Omega(t_0)$ by $\Omega = R^T\dot{R}$, then find $\omega(t_0)$,
2. Solve equation (5-23) numerically to obtain $\omega(t)$, and then $\Omega(t)$,
3. Solve the matrix differential equation $\dot{R} = R\Omega$ numerically to obtain $R(t)$. The solution $R(t)$ must be a rotation matrix,
4. Obtain the numerical value of $P(t)$ from equation (5-24),
5. Since $P(t)$ is given by $P = R^T\left(\ddot{T}_I - a_g\right)$, the linear displacement $T_I(t)$ can be obtained by utilizing initial conditions $T_I(t_0), \dot{T}_I(t_0)$, and doubly integrating $\ddot{T}_I(t) = R(t)P(t) + a_g$.

5.1.2 Comparing error of GF-INS and conventional INS

In section 2.1.4, the principle of conventional INS is shown. Three accelerometers sense the linear accelerations while three gyroscopes sense the angular velocities. The sensed velocities are firstly integrated to provide the angles turned. The error of these angles grows linear with time t. Then the linear accelerations are transformed with respect to the inertial frame utilizing the calculated angles. After double integration, the error rate of the determined position is in order of t^3.

On the other hand, GF-INS contains only linear accelerometers. The angular accelerations are calculated from the linear combination of the sensing of the six accelerometers. Obviously, the error rate of the calculated angles turned is in order of t^2. Besides, the linear accelerations involve not only the linear combination of the six acceleration data, but also the product of corresponding angular velocities. Therefore, the error rate of the calculated position is in order of t^4.

The error rate of GF-INS is an order higher than that of conventional INS. So it is important to have error reduction methods to enhance the performance of GF-INS.

5.1.3 Simulation study

To study the principle of the GF-INS and the algorithm proposed by Tan and Park, two simulations are performed. Two simulations are similar. The center of the rigid body is initially at $[10 \quad 0 \quad 0]^T$ with respect to the inertial frame. The rigid body rotates about both z-axes of both inertial frame and body frame. The difference is the first one is in constant angular velocity while the second one is in constant angular acceleration.

I. Simulation 1

The rigid body rotates with constant angular velocities $1\ rad/s$ as shown below,

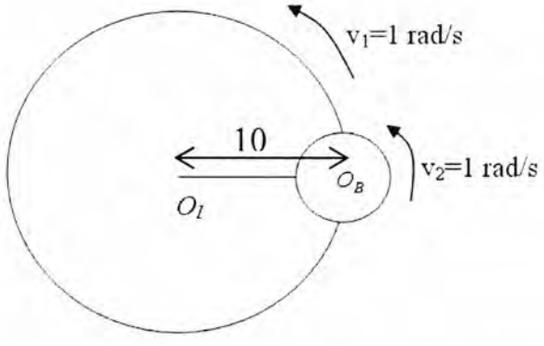


Figure 5-3 Simulated motion

For the rotation of the rigid body about the z-axis of the body frame, the rotation matrix

$$R(t) = \begin{bmatrix} \cos(v_2 t) & -\sin(v_2 t) & 0 \\ \sin(v_2 t) & \cos(v_2 t) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\Omega(t) = R(t)^T \dot{R}(t) = v_2 \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

There is another motion that rotates about the z-axis of the inertia frame. As the body frame does not rotate with respect to the inertial frame, it is only a translation of the rigid body with respect to the inertial frame. So,

$$T_I(t) = 10[\cos(v_1 t)x_1 + \sin(v_1 t)y_1]$$

$$\dot{T}_I(t) = 10v_1[-\sin(v_1 t)x_1 + \cos(v_1 t)y_1]$$

$$\ddot{T}_I(t) = -10v_1^2[\cos(v_1 t)x_1 + \sin(v_1 t)y_1]$$

Then

$$P = R^T(\ddot{T}_I - a_g) = \begin{bmatrix} -\delta v_1^2[\cos(v_2 t)\cos(v_1 t) + \sin(v_2 t)\sin(v_1 t)] \\ -\delta v_1^2[-\sin(v_2 t)\cos(v_1 t) + \cos(v_2 t)\sin(v_1 t)] \\ -g \end{bmatrix}$$

With sampling time 0.01s, the readings of the accelerometers are:

Accelerometer Readings	t = 0	t = 0.01	..	t = 10
A1	-7.0711	-7.0711	..	-7.0711
A2	-0.13435	-0.13435	..	-0.13435
A3	6.9367	6.9367	..	6.9367
A4	6.9367	6.9367	..	6.9367
A5	14.008	14.008	..	14.008
A6	7.0711	7.0711	..	7.0711

Table 5-1 Simulated readings of accelerometers

These data and the initial conditions are utilized to calculate the linear and angular motion of the rigid body. The detail procedures are listed below:

1) From equation (5-23), obtain the value of $\dot{\omega}(t)$,
2) Integrate $\dot{\omega}(t)$ to obtain $\omega(t)$ with $\omega(0) = [0 \quad 0 \quad 1]^T$,
3) Obtain $\Omega(t)$,
4) Solve $\dot{R} = R\Omega$, as Ω is a constant in this case, $F(t) = F(0)e^{\Omega t}$,
5) From equation (5-24), obtain $P(t)$,
6) Obtain $\ddot{T}_I(t)$ utilizing the relation $\ddot{T}_I(t) = R(t)P(t) + a_g$

7) Double integrate $\ddot{T}_I(t)$ to obtain $T_I(t)$ with $T_I(0) = [10 \quad 0 \quad 0]^T$ and $\dot{T}_I(0) = [0 \quad 10 \quad 0]^T$

8) Find the angle turned and the displacement translated from $R(t)$ and $T(t)$ respectively. The traveled trajectory is shown in Figure 5-4.

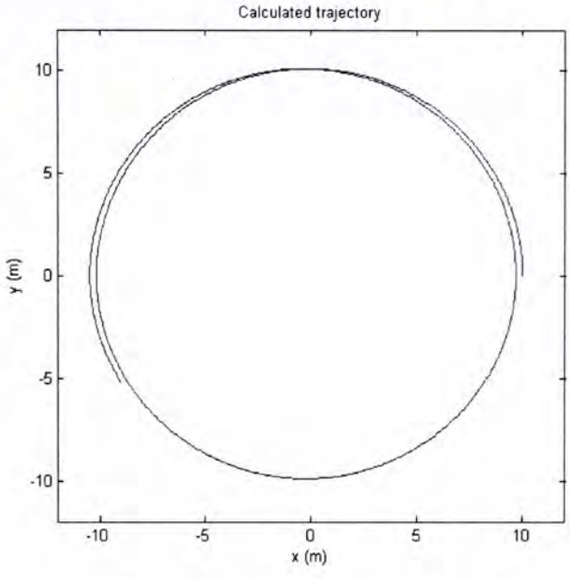


Figure 5-4 Calculated trajectory

II. Simulation 2

It is similar to the first simulation but with constant angular acceleration $1\ rad/s^2$ instead of angular velocity.

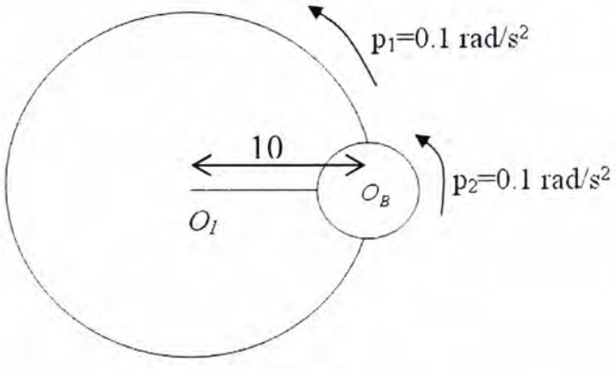


Figure 5-5 Simulated motion

For the rotation of the rigid body about the z-axis of the body frame,

$$\Omega(t) = t\Omega_z = 0.1t\begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$G(t) = \dot{\Omega}(t) + \Omega(t)^2 = \begin{bmatrix} -0.01t^2 & -0.1 & 0 \\ 0.1 & -0.01t^2 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

the rotation matrix

$$R(t) = R(t_0)e^{\theta(t)\Omega_0} = e^{0.05t^2\begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}}$$

$$R(t) = \begin{bmatrix} \cos(0.05t^2) & -\sin(0.05t^2) & 0 \\ \sin(0.05t^2) & \cos(0.05t^2) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

For the translation of the rigid body with respect to the inertial frame,

$$T_I(t) = 10[\cos(0.05t^2)x_1 + \sin(0.05t^2)y_1]$$

$$\dot{T}_I(t) = t[-\sin(0.05t^2)x_1 + \cos(0.05t^2)y_1]$$

$$\ddot{T}_I(t) = \begin{bmatrix} -0.1t^2(\cos(0.05t^2)) - \sin(0.05t^2) \\ -0.1t^2(\sin(0.05t^2)) + \cos(0.05t^2) \\ 0 \end{bmatrix}$$

Then

$$P(t) = R^T(\ddot{T}_I - a_g)$$

$$P(t) = \begin{bmatrix} \cos(0.05t^2) & -\sin(0.05t^2) & 0 \\ \sin(0.05t^2) & \cos(0.05t^2) & 0 \\ 0 & 0 & 1 \end{bmatrix}^T \begin{bmatrix} -0.1t^2(\cos(0.05t^2)) - \sin(0.05t^2) \\ -0.1t^2(\sin(0.05t^2)) + \cos(0.05t^2) \\ -g \end{bmatrix}$$

With sampling time 0.01s, the readings of the accelerometers are:

Accelerometer Readings	t = 0	t = 0.01	..	t = 10	..	t = 20
A1	0.7071	0.7071	..	-6.3428	..	-27.5489
A2	6.9405	6.9405	..	-0.1093	..	-21.3155
A3	7.6400	7.6400	..	7.6400	..	7.6400
A4	6.2258	6.2258	..	6.2258	..	6.2258
A5	6.9405	6.9405	..	13.9904	..	35.1965
A6	0.7071	0.7071	..	7.7570	..	28.9631

Table 5-2 Simulated readings of accelerometers

Then these data and the initial conditions are utilized to calculate the linear and angular motion of the rigid body. The detail procedures are listed below:

1) From equation (5-23), obtain the value of $\dot{\omega}(t)$,
2) Integrate $\dot{\omega}(t)$ to obtain $\omega(t)$ with $\omega(0) = [0 \quad 0 \quad 0]^T$,
3) Obtain $\Omega(t)$,
4) Solve $\dot{R} = R\Omega$, by using the approximation $\frac{\Delta R}{\Delta t} = R\Omega(t)$,
5) From equation (5-24) , obtain $P(t)$,
6) Obtain $\ddot{T}_I(t)$ utilizing the relation $\ddot{T}_I(t) = R(t)P(t) + a_g$,
7) Double integrate $\ddot{T}_I(t)$ to obtain $T_I(t)$ with $T_I(0) = [10 \quad 0 \quad 0]^T$ and $\dot{T}_I(0) = [0 \quad 0 \quad 0]^T$
8) Find the angle turned and the displacement translated from $R(t)$ and $T(t)$ respectively.

The result is shown in Figure 5-6.

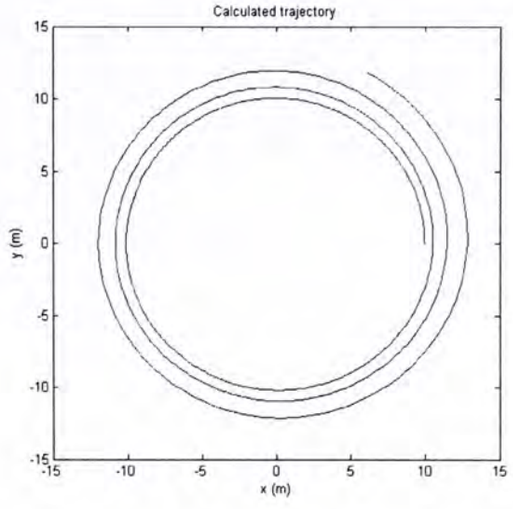


Figure 5-6 Calculated trajectory

The result tracks the real circular path nicely. However, the error is shown to be increasing with time. The errors compared with the real path are shown as follow.

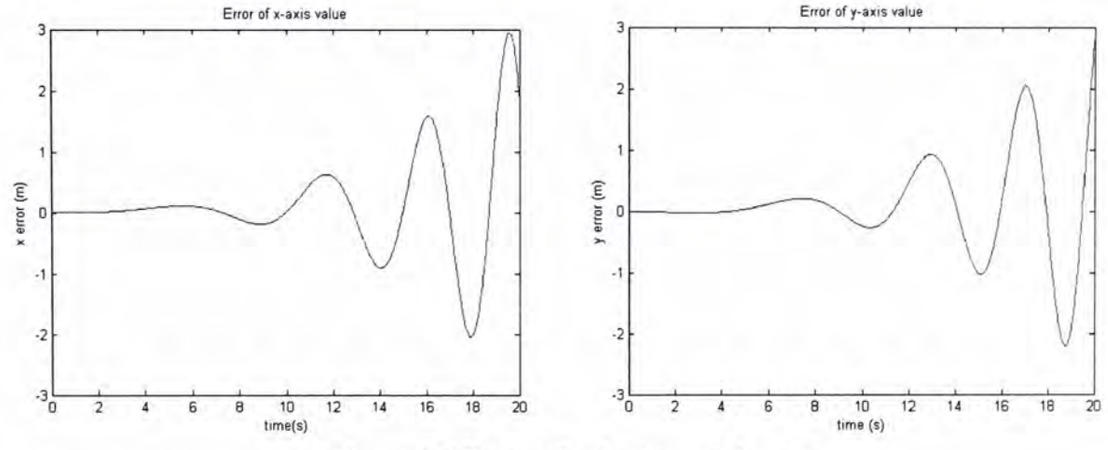


Figure 5-7 Errors of value on x and y axes

These two simulations illustrate the principle of a GF-INS. Although the readings of the accelerometers are perfect, there are still errors on the calculated displacement. This is because of the integration drift in numerical integration. One solution is to increase the sampling frequency. From equation (3-4), a smaller sampling time interval can increase the accuracy of the numerical integration.

Moreover, the errors of GF-INS are growing with time. It shows that it is only suitable for a short period of time in position determination unless there is calibration on position from external source.

5.2 Experimental setup

In chapters three and four, we have shown that the performance of MEMS accelerometer ADXL103 is capable for position determination. Therefore, the accelerometer is utilized again to construct a GF-INS. This time, six ADXL103 are utilized and placed in a particular configuration proposed by Tan and Park. The one dimensional INS presented in chapter three is modified. The circuit designed for each accelerometer is shown in Figure 5-8.

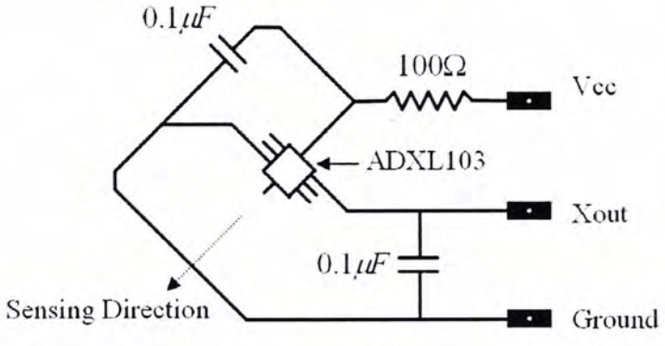


Figure 5-8 Circuit diagram of each side of the cube

Each side of the cube contains one accelerometer with the above circuit. The size of the cube is about $10cm \times 10cm \times 10cm$. Figure 5-9 shows the sensor part of the GF-INS.

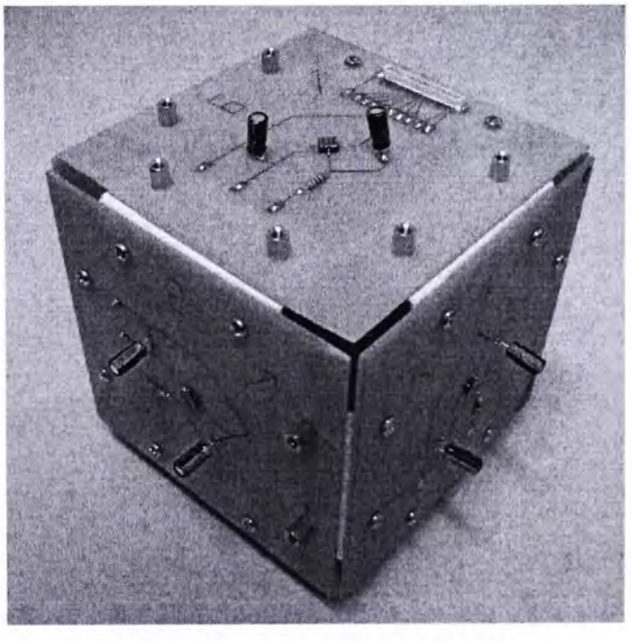


Figure 5-9 Sensor part of GF-INS

The cube contains a 51-pins socket which is able to connect to a MICA Mote. The architecture of the GF-INS is similar to the one dimensional INS which is presented in chapter 3. It is shown in the figure below.

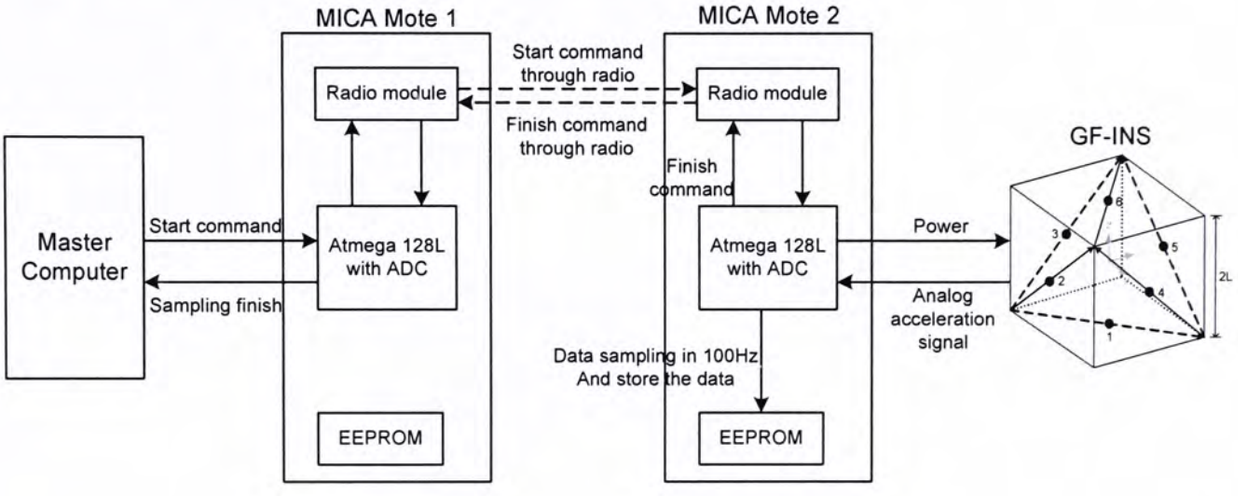


Figure 5-10 Architecture of the GF-INS

The six accelerometers are connected to six channels of the ADC in MICA Mote 2. When sampling signal is received by the microprocessor, the readings on the ADCs are corrected sequentially and then stored in the flash memory. The whole system is shown below.

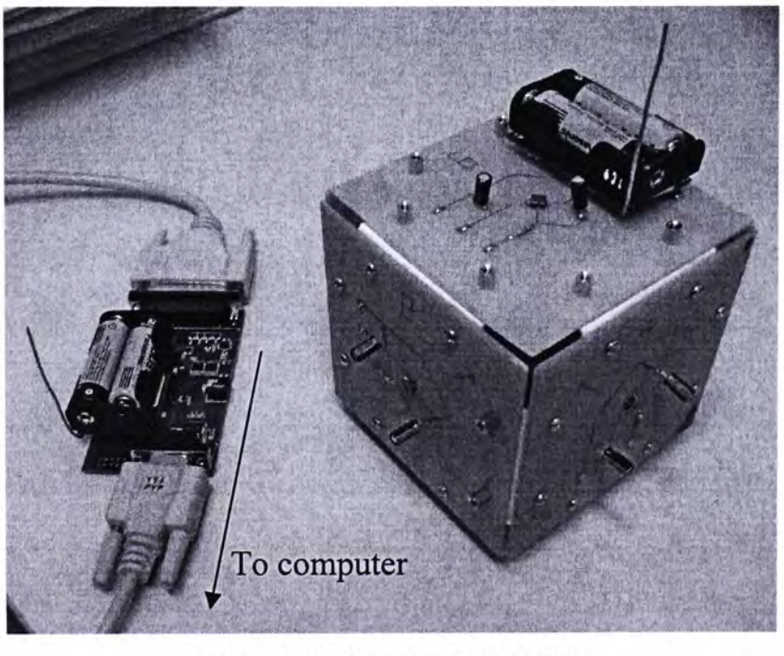


Figure 5-11 Overview of GF-INS

5.3 Experimental Results

The testing platform is the general-purpose industrial robot with five DOF which is described in section 3.3.4. The GF-INS sensor box is mounted on the tool end of the robot, as shown in Figure 5-12. The body frame on the sensor box is set as same as the Cartesian coordinate frame of the robot. Both linear and angular distance determinations are tested.

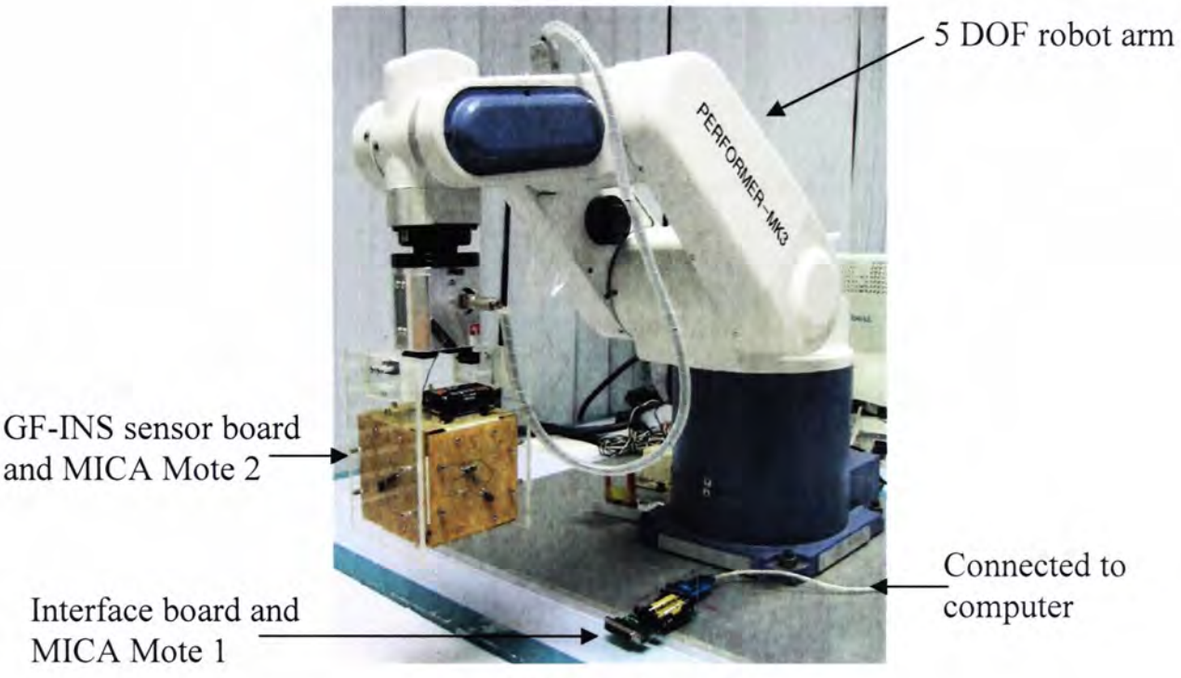


Figure 5-12 Experimental setup

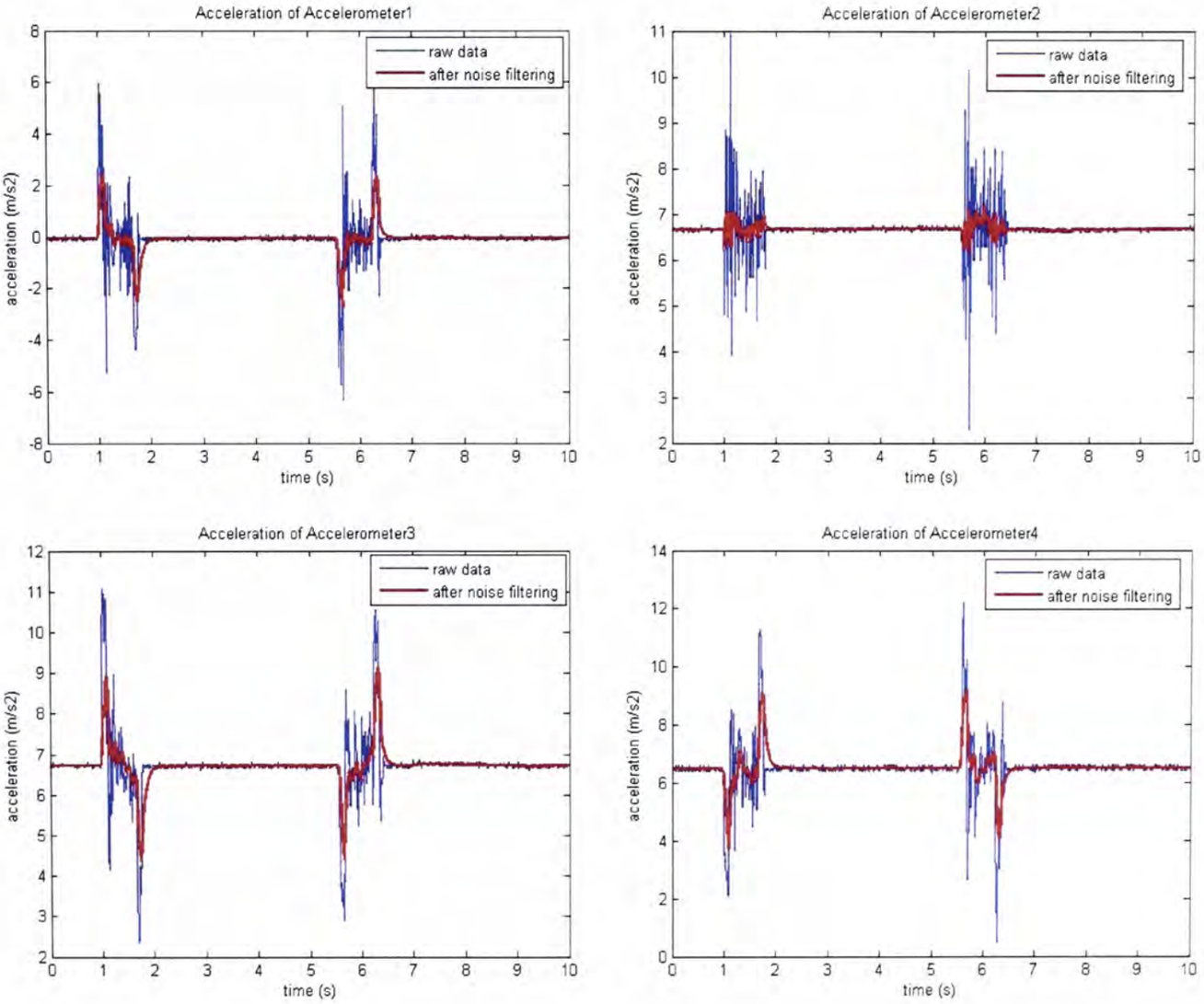
First of all, calibrations are needed to convert the ADC reading to meaningful acceleration data. The calibrations are made use of gravity and similar to that in section 3.4.1. The results are as follow.

Accelerometer	a_{+1g}	a_{0g}	a_{-1g}
A1	343	539	741
A2	340	534	742
A3	340	547	734
A4	671	782	889
A5	664	772	879
A6	328	529	732

The combination of the Kalman filter and the fuzzy logic based steady state detector described in chapter four are utilized here to filter the random noise and the small vibration of the acceleration data when in steady state. The principle of combining these two methods is discussed in section 4.2.2.

I. Experiment 1

In the first test, the mobile unit is moving in a return trip along y-axis of the inertial frame with a distance 0.257m. The sensed accelerations of the six accelerometers are shown below.



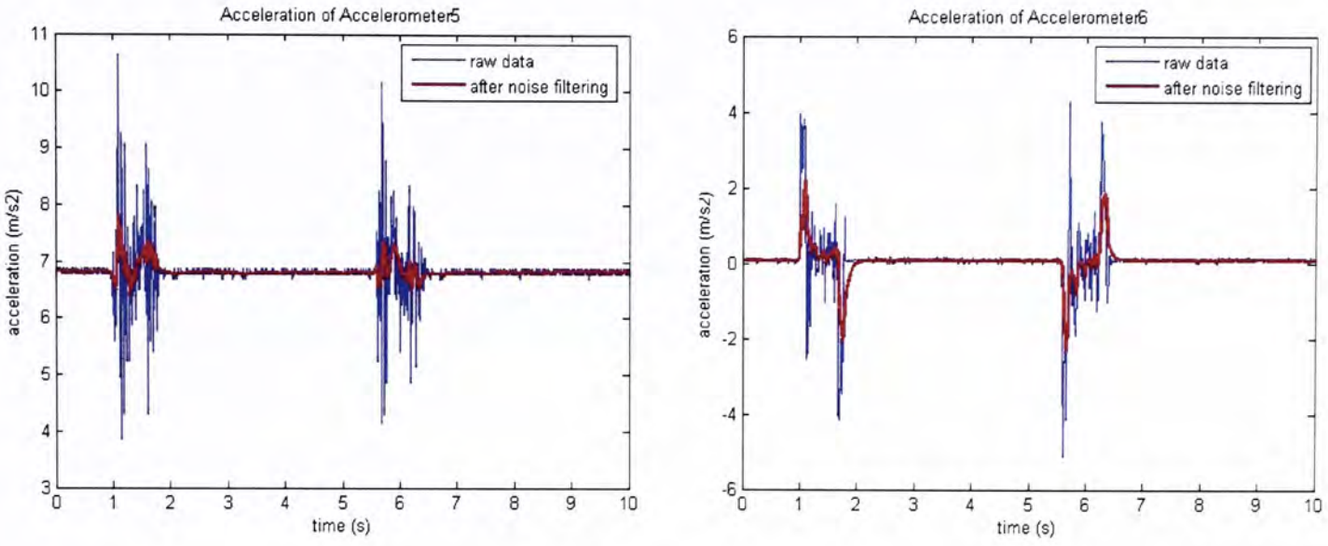


Figure 5-13 Experimental data

To calculate the displacement, equations (5-23) and (5-24) are utilized. The calculated results are as follow.

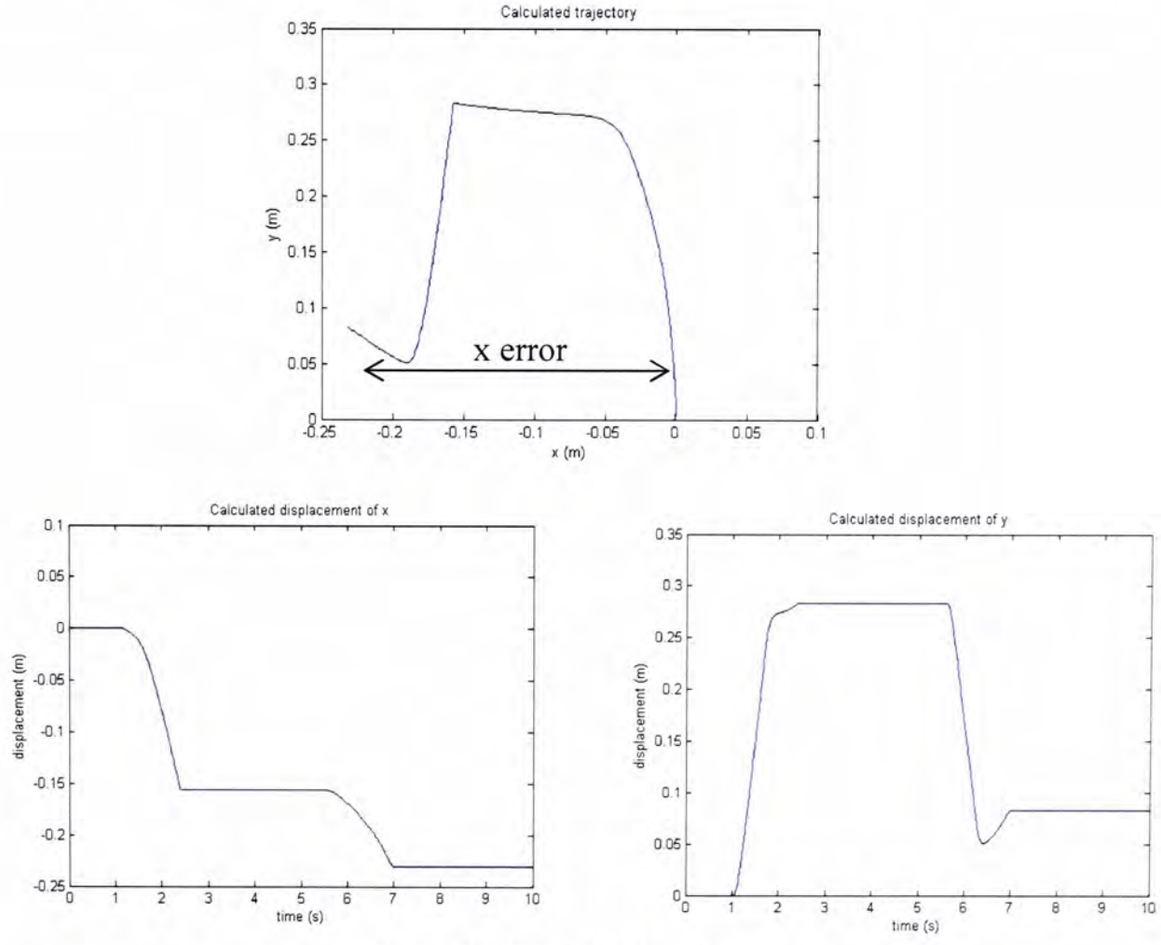
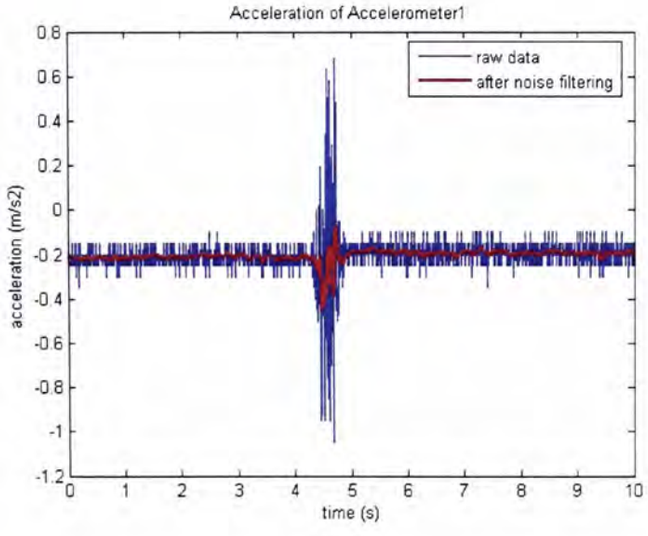


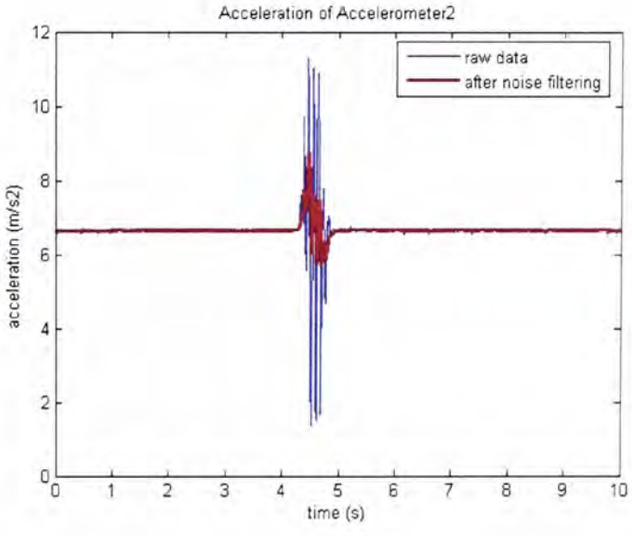
Figure 5-14 Calculated results

In this test, the unit does not move along x-axis. However, the calculated results show that there is movement on x-axis direction. It is because the algorithm of GF-INS involves linear combination of the six accelerometers. Theoretically, the first component of P in equation (5-24) should remain zero if there is no movement along x-axis of the body frame. But it is not that perfect in real case. Due to the slightly inaccurate of the acceleration measurements on the six accelerometers, the linear combination of the readings is difficult to remain zero throughout the testing. It is one of the challenges to implement the GF-INS in real case. However, the algorithm works in determining the movement along y-axis. So once the error of the GF-INS can be reduced, it still has the potential to implement in real application.

II. Experiment 2

The second test is about the angular movement. The mobile unit rotates about z-axis of the body frame with 90 degrees. The sensed accelerations are shown below.





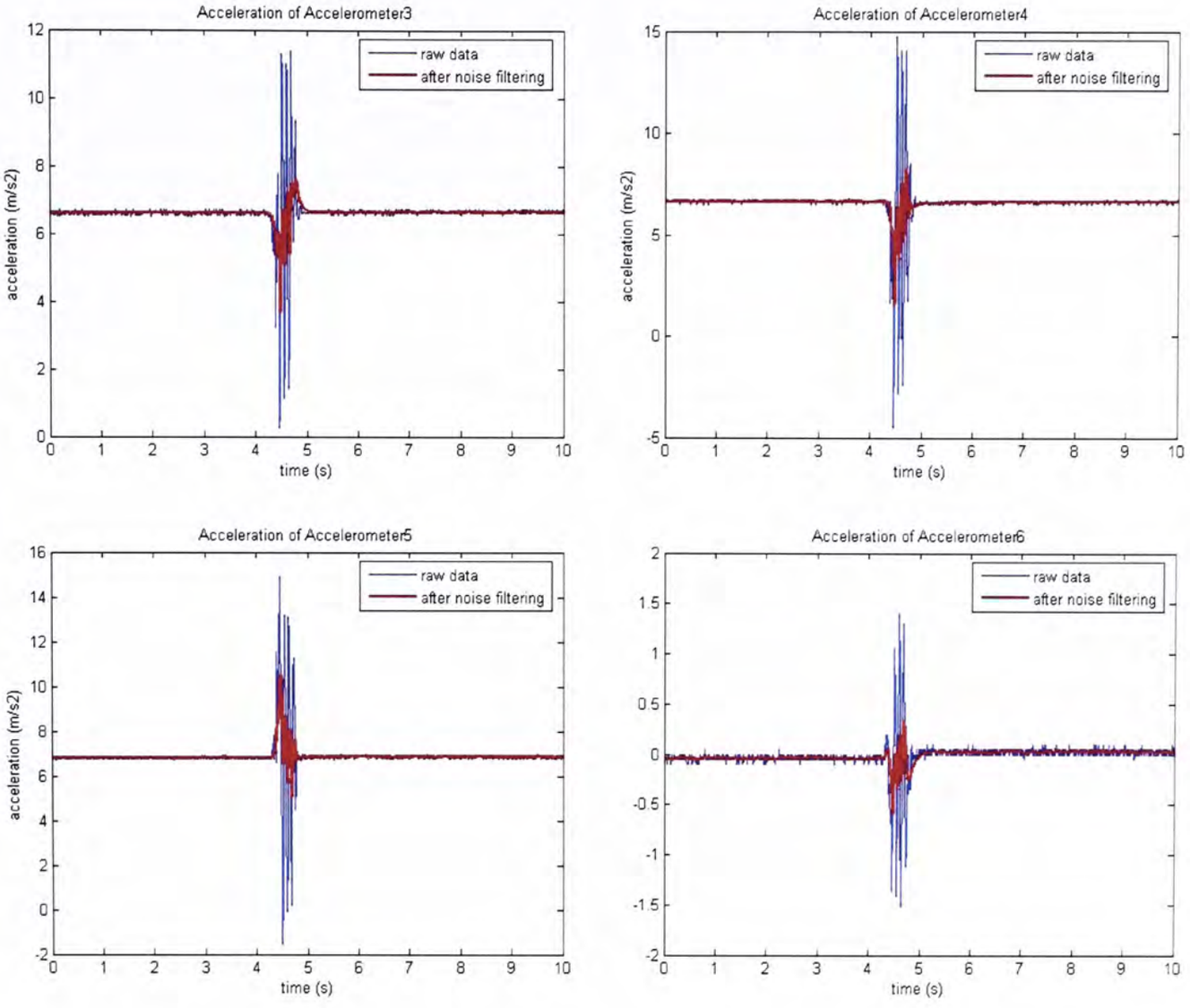


Figure 5-15 Experimental data

The accelerations are processed to calculate the movement and the angle turned. The results are shown as follow.

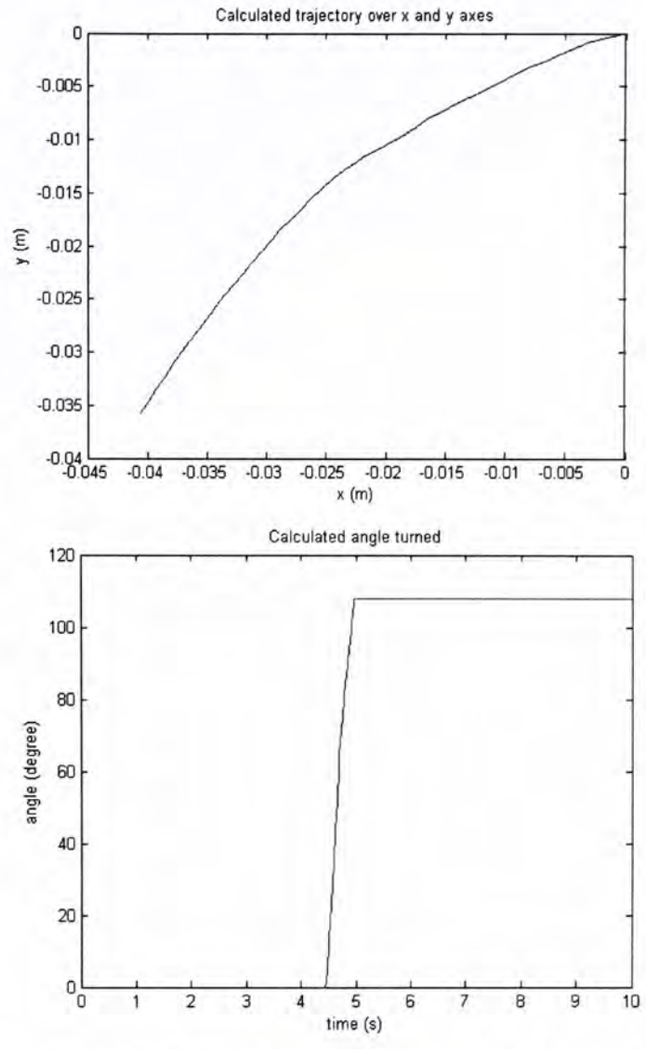


Figure 5-16 Calculated trajectory and angle turned

In this test, the movement of any direction should be zero theoretically. However, there are calculated movement along both x and y axes. The situation is similar to the previous test. It is because of the slightly inaccurate accelerations data. But if we focus on the detected angle turned, the performance of the GF-INS is reasonable and acceptable.

These two tests illustrate the sensitivity of the GF-INS algorithm. Since the calculation depends on the linear combinations of the six accelerations readings, the requirement of the accuracy of the data is very high. In other words, extra work should be done on enhancing the accuracy. This is the aim of the next chapter.

5.4 Summary

In chapter four, the performance of the MEMS accelerometer on distance determination is shown to be acceptable. So we extend the MEMS accelerometer based one dimensional INS to GF-INS and the details are shown in this chapter.

Utilizing linear accelerometer to determine both linear and angular motion of rigid body has been discussed for forty years [49]. With the efforts of numerous researchers, it is proved that only six linear accelerometers are enough to determine the motion of the rigid body. Our GF-INS is based on the algorithm developed by Tan and Park [45]. They proposed a particular cube-shape configuration with six accelerometers to form a GF-INS. Their algorithm is briefly introduced in this chapter. Two simulations studies are carried out to familiarize the algorithm.

The hardware of the GF-INS is similar to the one dimensional INS discussed in chapter three. The sensor broad is replaced by a cube shape sensor box with six accelerometers which are located on the center of diagonal of each side.

The setup is tested with the robot arm platform. Both linear and angular motions are tested. The results show that the GF-INS is sensitive with noise. This is because the algorithm depends on the linear combination of the six accelerometers. With small inaccurate accelerations, the error is large due to the double integrations. On the other hand, the results on the moving axes are still reasonable. This implies that GF-INS has the potential in position determination. The challenge is to reduce the calculated error along the axes with no movement.

Chapter 6
Improvement on the GF-INS

To increase the accuracy and practical feasibility of GF-INS, two methods are proposed in this chapter. The first one is the configuration error compensation which is originally developed by Tan and Park [45]. We modify it to suit the hardware in our case. The second one is a fuzzy rule based motion state detector. With this, the algorithm of GF-INS can be simplified when the state of motion is known.

6.1 Configuration error compensation

The errors of GF-INS can be categorized into three types: accelerometer sensor errors, configuration error and computational process errors. Accelerometer sensor error includes bias error, scale factor error and error from random noise, modeled as equation (3-13). Kalman filtering described in section 4.2 is a method to reduce the error from random noise. Configuration error is due to the misalignment between the ideal and the real position and sensing direction of the inertial sensors. The typical computational process error is the numerical integration error. The error can be reduced by increasing the sampling frequency.

Besides the algorithm of GF-INS, Tan and Park [45] also propose the methods to identify the bias error, scale factor error, position error and sensing direction error. However, because of the hardware limitation of the testing platform, their method in finding the position error is not practical in our system. The methods are modified to make them more practical and general.

6.1.1 Identify bias, scale factor and sensing direction error

The output of the accelerometer of each side of the device can be modeled as

$$A_{ei} = (1 + s_i)A_{ri} + b_i \tag{6-1}$$

Where A_{ei} is the accelerometer reading with errors, A_{ri} is the reading with real position and sensing direction of the accelerometers. s_i is the scale factor error and b_i is the bias error.

Assume the real position is

$$u_{ri} = L\begin{bmatrix} u_{r1x} & u_{r2x} & u_{r3x} & u_{r4x} & u_{r5x} & u_{r6x} \\ u_{r1y} & u_{r2y} & u_{r3y} & u_{r4y} & u_{r5y} & u_{r6y} \\ u_{r1z} & u_{r2z} & u_{r3z} & u_{r4z} & u_{r5z} & u_{r6z} \end{bmatrix} \tag{6-2}$$

And the real sensing direction is

$$\theta_{ri} = \begin{bmatrix} \theta_{r1x} & \theta_{r2x} & \theta_{r3x} & \theta_{r4x} & \theta_{r5x} & \theta_{r6x} \\ \theta_{r1y} & \theta_{r2y} & \theta_{r3y} & \theta_{r4y} & \theta_{r5y} & \theta_{r6y} \\ \theta_{r1z} & \theta_{r2z} & \theta_{r3z} & \theta_{r4z} & \theta_{r5z} & \theta_{r6z} \end{bmatrix} \tag{6-3}$$

Consider equation (5-11),

$$\begin{aligned} A_i &= \left\langle R^T\left(\ddot{T}_I - a_g\right) + \left(\dot{\Omega} + \Omega^2\right)u_i, \theta_i \right\rangle \\ &= \theta_i^T R^T\left(\ddot{T}_I - a_g\right) + \theta_i^T\left(\dot{\Omega} + \Omega^2\right)u_i \end{aligned} \tag{6-4}$$

If the rigid body is in steady state, $\Omega = 0$ and $\ddot{T}_I = 0$, then

$$A_i = \theta_i^T R^T\left(-a_g\right) \tag{6-5}$$

It is found that if the rigid body is in steady state, the positions of the accelerometers would not affect the acceleration readings. That implies the sensing direction error of the accelerometers can be identified.

From [45], six sets of data in steady state are taken.

(1) Firstly, collect the data with z-axis aligning negative gravity. Then

$$A_{ei}^{(1)} = (1+s_i)a_g[-\theta_{r1z} \quad -\theta_{r2z} \quad -\theta_{r3z} \quad -\theta_{r4z} \quad -\theta_{r5z} \quad -\theta_{r6z}]^T + b_i$$

(2) Then rotate along y axis by -180 degree, collect the data with z-axis aligning positive gravity

$$A_{ei}^{(2)} = (1+s_i)a_g[\theta_{r1z} \quad \theta_{r2z} \quad \theta_{r3z} \quad \theta_{r4z} \quad \theta_{r5z} \quad \theta_{r6z}]^T + b_i$$

(3) Rotate along y axis by -90 degree, collect the data with x-axis aligning negative gravity

$$A_{ei}^{(3)} = (1+s_i)a_g[-\theta_{r1x} \quad -\theta_{r2x} \quad -\theta_{r3x} \quad -\theta_{r4x} \quad -\theta_{r5x} \quad -\theta_{r6x}]^T + b_i$$

(4) Rotate along z axis by 180 degree and then y axis by 90 degree, collect the data with x axis aligning positive gravity

$$A_{ei}^{(4)} = (1+s_i)a_g[\theta_{r1x} \quad \theta_{r2x} \quad \theta_{r3x} \quad \theta_{r4x} \quad \theta_{r5x} \quad \theta_{r6x}]^T + b_i$$

(5) Rotate along z axis by -90 degree and then x axis by 90 degree, collect the data with y axis aligning negative gravity

$$A_{ei}^{(5)} = (1+s_i)a_g[-\theta_{r1y} \quad -\theta_{r2y} \quad -\theta_{r3y} \quad -\theta_{r4y} \quad -\theta_{r5y} \quad -\theta_{r6y}]^T + b_i$$

(6) Rotate along z axis by 90 degree and then x axis by -90 degree, collect the data with y axis aligning positive gravity

$$A_{ei}^{(6)} = (1+s_i)a_g[\theta_{r1y} \quad \theta_{r2y} \quad \theta_{r3y} \quad \theta_{r4y} \quad \theta_{r5y} \quad \theta_{r6y}]^T + b_i$$

Then the bias error can be obtained by,

$$A_{ei}^{(1)} + A_{ei}^{(2)} + A_{ei}^{(3)} + A_{ei}^{(4)} + A_{ei}^{(5)} + A_{ei}^{(6)} = 6b_i$$

$$b_i = \frac{A_{ei}^{(1)} + A_{ei}^{(2)} + A_{ei}^{(3)} + A_{ei}^{(4)} + A_{ei}^{(5)} + A_{ei}^{(6)}}{6} \tag{6-6}$$

To find the real sensing direction,

$$A_{ei}^{(2)} - A_{ei}^{(1)} = 2(1+s_i)a_g\begin{bmatrix}\theta_{r1z} & \theta_{r2z} & \theta_{r3z} & \theta_{r4z} & \theta_{r5z} & \theta_{r6z}\end{bmatrix}^T$$

$$\theta_{riz} = \frac{A_{ei}^{(2)} - A_{ei}^{(1)}}{2(1+s_i)a_g}$$

$$A_{ei}^{(4)} - A_{ei}^{(3)} = 2(1+s_i)a_g\begin{bmatrix}\theta_{r1x} & \theta_{r2x} & \theta_{r3x} & \theta_{r4x} & \theta_{r5x} & \theta_{r6x}\end{bmatrix}^T$$

$$\theta_{rix} = \frac{A_{ei}^{(4)} - A_{ei}^{(3)}}{2(1+s_i)a_g}$$

$$A_{ei}^{(6)} - A_{ei}^{(5)} = 2(1+s_i)a_g\begin{bmatrix}\theta_{r1y} & \theta_{r2y} & \theta_{r3y} & \theta_{r4y} & \theta_{r5y} & \theta_{r6y}\end{bmatrix}^T$$

$$\theta_{riy} = \frac{A_{ei}^{(6)} - A_{ei}^{(5)}}{2(1+s_i)a_g}$$

Therefore,

$$\begin{bmatrix}\theta_{rix}\\ \theta_{riy}\\ \theta_{riz}\end{bmatrix} = \frac{1}{2(1+s_i)a_g}\begin{bmatrix}A_{ei}^{(4)} - A_{ei}^{(3)}\\ A_{ei}^{(6)} - A_{ei}^{(5)}\\ A_{ei}^{(2)} - A_{ei}^{(1)}\end{bmatrix} \tag{6-7}$$

Taking i=1 as an example,

$$\begin{bmatrix}\theta_{r1x}\\ \theta_{r1y}\\ \theta_{r1z}\end{bmatrix} = \frac{1}{2(1+s_1)a_g}\begin{bmatrix}A_{e1}^{(4)} - A_{e1}^{(3)}\\ A_{e1}^{(6)} - A_{e1}^{(5)}\\ A_{e1}^{(2)} - A_{e1}^{(1)}\end{bmatrix}$$

As the sensing direction is unit vector,

$$\begin{bmatrix}\theta_{r1x}\\ \theta_{r1y}\\ \theta_{r1z}\end{bmatrix} = -\begin{bmatrix}A_{e1}^{(4)} - A_{e1}^{(3)}\\ A_{e1}^{(6)} - A_{e1}^{(5)}\\ A_{e1}^{(2)} - A_{e1}^{(1)}\end{bmatrix} \Bigg/ \begin{vmatrix}A_{e1}^{(4)} - A_{e1}^{(3)}\\ A_{e1}^{(6)} - A_{e1}^{(5)}\\ A_{e1}^{(2)} - A_{e1}^{(1)}\end{vmatrix}$$

With six sets of data, the 18 components of real sensing direction θ_{ri} can be identified. Put the values back to the equation (6-7), the scale factor error of each accelerometer can be found.

6.1.2 Identify position error

From equation (6-4), the positions of the accelerometers affect the performance only when the rigid rotates. So for simplicity, the rigid body is rotated at constant velocity and three sets of data are collected. In the method of Tan and Park, the cube is placed on an inclined plane and the rotation axis should pass through the center of the cube. It is not practical in our testing platform. So the method is modified and as shown below.

The three sets data are,

(1) Rotate about z axis which is parallel to the gravity with constant angular velocity $1 rad/s$,

$$\omega^{(1)} = [0 \quad 0 \quad 1]^T$$

$$\Omega^{(1)} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$R^{(1)}(t) = \begin{bmatrix} \cos t & -\sin t & 0 \\ \sin t & \cos t & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$A_{ri}^{(1)} = \theta_{ri}^T R^{(1)^T} \left(\ddot{T}_1 - a_g\right) + \theta_{ri}^T \Omega^{(1)^2} u_{ri}$$

The real acceleration is

$$A_{ei}^{(1)} = (1+s_i)\left[\theta_{ri}^T R^{(1)^T} \left(\ddot{T}_1 - a_g\right) + \theta_{ri}^T \Omega^{(1)^2} u_{ri}\right] + b_i$$

$$\frac{A_{ei}^{(1)} - b_i}{1+s_i} = \theta_{ri}^T R^{(1)^T} \left(\ddot{T}_1 - a_g\right) + \theta_{ri}^T \Omega^{(1)^2} u_{ri}$$

Then we have the following six equations:

$$\frac{A_{ei}^{(1)} - b_i}{1+s_i} = \theta_{ri}^T [0 \quad 0 \quad -a_g]^T + \theta_{ri}^T [-u_{rix} \quad -u_{riy} \quad 0]^T \tag{6-8}$$

(2) Rotate about x axis which is parallel to the gravity with constant angular velocity $1 rad/s$,

For rotation part,

$$\omega^{(2)} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T$$

$$\Omega^{(2)} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$R^{(2)}(t) = \begin{bmatrix} 0 & -\sin t & -\cos t \\ 0 & \cos t & -\sin t \\ 1 & 0 & 0 \end{bmatrix}$$

Assume α is the distance between the center of cube and the rotation center. Then the translation part,

$$\ddot{T}_I(t) = -\alpha \begin{bmatrix} \cos t \\ \sin t \\ 0 \end{bmatrix}$$

Then

$$P = R^T(\ddot{T}_I - a_g) = \begin{bmatrix} -a_g \\ 0 \\ \alpha \end{bmatrix}$$

$$A_{ri}^{(2)} = \theta_{ri}^T R^{(2)^T}\left(\ddot{T}_I - a_g\right) + \theta_{ri}^T \Omega^{(2)^2} u_{ri}$$

The actual acceleration is,

$$A_{ei}^{(2)} = (1+s_i)\left[\theta_{ri}^T R^{(2)^T}\left(\ddot{T}_I - a_g\right) + \theta_{ri}^T \Omega^{(2)^2} u_{ri}\right] + b_i$$

$$\frac{A_{ei}^{(2)} - b_i}{1+s_i} = \theta_{ri}^T F^{(2)^T}\left(\ddot{R}_I - a_g\right) + \theta_{ri}^T \Omega^{(2)^2} u_{ri}$$

Then we have the following six equations:

$$\frac{A_{ei}^{(2)} - b_i}{1+s_i} = \theta_{ri}^T \begin{bmatrix} -a_g & 0 & \alpha \end{bmatrix}^T + \theta_{ri}^T \begin{bmatrix} 0 & -u_{riy} & -u_{riz} \end{bmatrix}^T \tag{6-9}$$

(3) Rotate about y axis which is parallel to the gravity with constant angular velocity $1 rad/s$,

For rotation part,

$$\omega^{(3)} = [0 \quad 1 \quad 0]^T$$

$$\Omega^{(3)} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix}$$

$$R^{(3)}(t) = \begin{bmatrix} \sin t & 0 & -\cos t \\ -\cos t & 0 & -\sin t \\ 0 & 1 & 0 \end{bmatrix}$$

Translation part:

$$\ddot{R}_I(t) = -\alpha \begin{bmatrix} \cos t \\ \sin t \\ 0 \end{bmatrix}$$

Then

$$P = R^T(\ddot{T}_I - a_g) = \begin{bmatrix} 0 \\ -a_g \\ \alpha \end{bmatrix}$$

$$A_{ri}^{(3)} = \theta_{ri}^T F^{(3)^T}\left(\ddot{R}_I - a_g\right) + \theta_{ri}^T \Omega^{(3)^2} u_{ri}$$

The actual acceleration is,

$$A_{ei}^{(3)} = (1+s_i)\left[\theta_{ri}^T F^{(3)^T}\left(\ddot{R}_I - a_g\right) + \theta_{ri}^T \Omega^{(3)^2} u_{ri}\right] + b_i$$

$$\frac{A_{ei}^{(3)} - b_i}{1+s_i} = \theta_{ri}^T F^{(3)^T}\left(\ddot{R}_I - a_g\right) + \theta_{ri}^T \Omega^{(3)^2} u_{ri}$$

Again we have the following six equations:

$$\frac{A_{ei}^{(3)} - b_i}{1+s_i} = \theta_{ri}^T [0 \quad -a_g \quad \alpha]^T + \theta_{ri}^T [-u_{rix} \quad 0 \quad -u_{riz}]^T \tag{6-10}$$

The system of equations (6-8), (6-9) and (6-10) contains 18 equations. It is sufficient to solve the 18 unknown components of the position matrix u_{ri}.

6.1.3 Compensator design

As the basic algorithm of GF-INS is based on the ideal position and the sensing direction of the accelerometers, the real data should be converted to the ideal one before inputted to the algorithm. Therefore, a compensator is designed to reduce the effect of the position and sensing direction error. Similar work can be found in [46]. The work presented here is the extent of [45]. The effect of the scale factor error and the bias error are added to improve the performance of the compensator.

The following shows the design and the procedures of utilizing the compensator.

Consider,

$$A_{ei} = (1+s_i)A_{ri} + b_i$$

$$A_{ei} = (1+s_i)\left(\theta_{ri}{}^T P + (u_{ri} \times \theta_{ri})^T \dot{\omega} + \begin{bmatrix} \theta_{r1}{}^T \Omega^2 u_{r1} \\ \vdots \\ \theta_{r6}{}^T \Omega^2 u_{r6} \end{bmatrix} \right) + b_i$$

$$A_{ei} = (1+s_i)\left(J_r \begin{bmatrix} \dot{\omega} \\ P \end{bmatrix} + \begin{bmatrix} \theta_{r1}{}^T \Omega^2 u_{r1} \\ \vdots \\ \theta_{r6}{}^T \Omega^2 u_{r6} \end{bmatrix} \right) + b_i \qquad (6\text{-}11)$$

Where $J_r = \left[(u_{ri} \times \theta_{ri})^T \quad \theta_{ri}{}^T \right]$

$$\text{As } A_i = \begin{bmatrix} A_1 \\ \vdots \\ A_6 \end{bmatrix} = J \begin{bmatrix} \dot{\omega} \\ P \end{bmatrix} + \begin{bmatrix} \theta_1{}^T \Omega^2 u_1 \\ \vdots \\ \theta_6{}^T \Omega^2 u_6 \end{bmatrix}$$

$$\begin{bmatrix} \dot{\omega} \\ P \end{bmatrix} = J^{-1}A_i - J^{-1}\begin{bmatrix} \theta_1{}^T\Omega^2 u_1 \\ \vdots \\ \theta_6{}^T\Omega^2 u_6 \end{bmatrix}$$

Where $J = \begin{bmatrix} (u_i \times \theta_i)^T & \theta_i{}^T \end{bmatrix}$

Therefore, equation (6-11) becomes,

$$A_{ei} = (1+s_i)\left(J_r\left(J^{-1}A_i - J^{-1}\begin{bmatrix} \theta_1{}^T\Omega^2 u_1 \\ \vdots \\ \theta_6{}^T\Omega^2 u_6 \end{bmatrix}\right) + \begin{bmatrix} \theta_{r1}{}^T\Omega^2 u_{r1} \\ \vdots \\ \theta_{r6}{}^T\Omega^2 u_{r6} \end{bmatrix}\right) + b_i$$

$$\frac{A_{ei} - b_i}{1+s_i} = J_r J^{-1}A_i - J_r J^{-1}\begin{bmatrix} \theta_1{}^T\Omega^2 u_1 \\ \vdots \\ \theta_6{}^T\Omega^2 u_6 \end{bmatrix} + \begin{bmatrix} \theta_{r1}{}^T\Omega^2 u_{r1} \\ \vdots \\ \theta_{r6}{}^T\Omega^2 u_{r6} \end{bmatrix}$$

$$A_i = \left(J_r J^{-1}\right)^{-1}\left[\frac{A_{ei} - b_i}{1+s_i} + J_r J^{-1}\begin{bmatrix} \theta_1{}^T\Omega^2 u_1 \\ \vdots \\ \theta_6{}^T\Omega^2 u_6 \end{bmatrix} - \begin{bmatrix} \theta_{r1}{}^T\Omega^2 u_{r1} \\ \vdots \\ \theta_{r6}{}^T\Omega^2 u_{r6} \end{bmatrix}\right] \quad (6\text{-}12)$$

From equation (6-12), the accelerations of ideal configuration can be obtained by utilizing the actual readings of accelerations, positions and sensing directions. The following shows the step of processing

(1) At time k, obtain the angular velocity $\omega_i(k)$ from $A_i(k)$

(2) Determine $\Omega_i(k)$

(3) Obtain $A_i(k+1)$ using $\Omega_i(k)$, A_{ei}, b_i, s_i, θ_{ri}, u_{ri}, θ_i and u_i

(4) Repeat step (1) until the end

6.1.4 Simulation

To illustrate the methodology of finding the scale factor error, bias error, position and sensing direction error, a simulation is performed. It also shows the effect of these errors and the performance of the compensator. It simulates a rigid body rotating about the z-axes of both body frame and inertial frame with constant angular velocity $1\ rad/s$. The motion is shown in Figure 6-1.

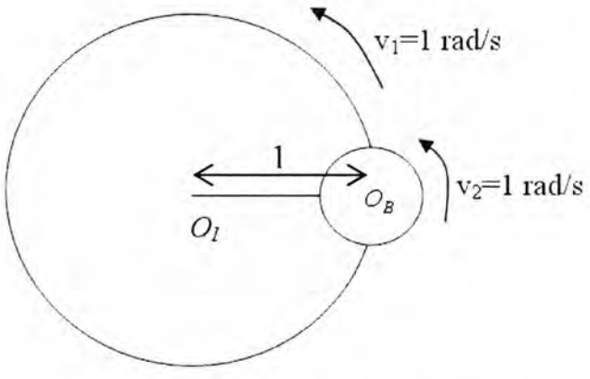


Figure 6-1 Simulated motion

In ideal case, the positions and the sensing directions of the accelerometers are the same as the particular configuration in Figure 5-2. And assume there are no scale factor error and bias error. The ideal readings of the accelerations are listed below.

Accelerometer Readings	t = 0	t = 0.01	..	t = 10
A1	-0.70711	-0.70711	..	-0.70711
A2	6.2275	6.2275	..	6.2275
A3	6.9347	6.9347	..	6.9347
A4	6.9347	6.9347	..	6.9347
A5	7.6418	7.6418	..	7.6418
A6	0.70711	0.70711	..	0.70711

Table 6-1 Ideal readings of accelerations

Assume there are bias error, scale factor error, position error and sensing direction error, the actual values are as follow.

Real position

Take $L = 0.054m$

$$u_{ri} = \begin{bmatrix} 0.0054 & 0.0054 & -0.0594 & 0.0594 & 0.0054 & 0.0054 \\ 0.0054 & -0.0594 & 0.0054 & 0.0054 & 0.0486 & 0.0054 \\ -0.0486 & 0.0054 & -0.0054 & 0.0054 & 0.0054 & 0.0594 \end{bmatrix}$$

Real sensing direction

$$\theta_{ri} = \begin{bmatrix} 0.772 & 0.7357 & 0.07 & -0.07 & -0.6317 & -0.7056 \\ 0.6317 & 0.0613 & 0.772 & -0.6317 & 0.07 & 0.7056 \\ 0.07 & 0.6744 & 0.6317 & 0.772 & 0.772 & 0.0642 \end{bmatrix}$$

Bias of each accelerometer

$$b_i = [0.5 \quad 0.5 \quad 1 \quad 1.5 \quad 0.5 \quad 0.5]^T$$

Scale factor error of each accelerometer

$$s_i = [0.01 \quad 0.02 \quad 0.01 \quad 0.5 \quad 0.06 \quad 0.02]^T$$

Then with sampling time 0.01s, the actual sensed readings

Accelerometer Readings	t = 0	t = 0.01	..	t = 10
A1	0.40778	0.40778	..	0.40778
A2	6.4959	6.4959	..	6.4959
A3	7.186	7.186	..	7.186
A4	9.5318	9.5318	..	9.5318
A5	9.1954	9.1954	..	9.1954
A6	1.8615	1.8615	..	1.8615

Table 6-2 Actual readings of accelerations

Apply the algorithm of GF-INS presented in section 5.1.1, the calculated path of the rigid body motion is shown in Figure 6-2.

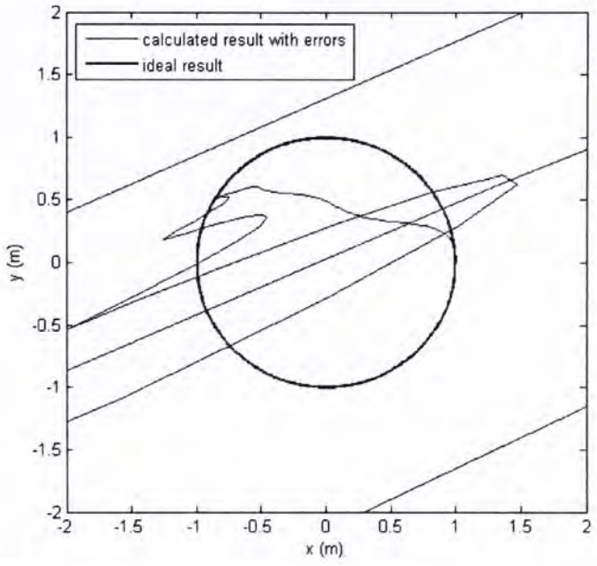


Figure 6-2 Calculated trajectory with actual readings and ideal readings

The result shows the effect of the errors on the performance of the GF-INS algorithm. It is because the algorithm is based on the assumption that the positions and sensing directions of the accelerometers are exactly the same as the ideal configuration. Therefore, it is necessary step to identify the configuration errors for the performance enhancement.

To identify the scale factor error, bias error and sensing direction error, the methods described in section 6.1.1 are performed. Below are the simulated values of the six tests.

	A_{e1}	A_{e2}	A_{e3}	A_{e4}	A_{e5}	A_{e6}
Test 1	1.1952	7.2467	7.2568	9.4501	8.5258	1.1417
Test 2	-0.19521	-6.2467	-5.2568	-6.4501	-7.5258	-0.14171
Test 3	8.1473	7.86	1.6952	0.77726	-6.0666	-6.5588
Test 4	-7.1473	-6.86	0.30479	2.2227	7.0666	7.5588
Test 5	6.7568	1.1133	8.6473	-5.0046	1.2296	7.5588
Test 6	-5.7568	-0.11334	-6.6473	8.0046	-0.22962	-6.5588

Table 6-3 Simulated values of the six stationary tests

From equation (6-6),

$$b_i = \frac{A_{ei}^{(1)} + A_{ei}^{(2)} + A_{ei}^{(3)} + A_{ei}^{(4)} + A_{ei}^{(5)} + A_{ei}^{(6)}}{6}$$

$$b_i = [0.5 \quad 0.5 \quad 1 \quad 1.5 \quad 0.5 \quad 0.5]^T$$

Then the actual sensing directions are

$$\begin{bmatrix} \theta_{rix} \\ \theta_{riy} \\ \theta_{riz} \end{bmatrix} = -\begin{bmatrix} A_{ei}^{(4)} - A_{ei}^{(3)} \\ A_{ei}^{(6)} - A_{ei}^{(5)} \\ A_{ei}^{(2)} - A_{ei}^{(1)} \end{bmatrix} \Bigg/ \left| \begin{matrix} A_{ei}^{(4)} - A_{ei}^{(3)} \\ A_{ei}^{(6)} - A_{ei}^{(5)} \\ A_{ei}^{(2)} - A_{ei}^{(1)} \end{matrix} \right|$$

$$\theta_{ri} = \begin{bmatrix} 0.7720 & 0.7358 & 0.0702 & -0.0702 & -0.6317 & -0.7057 \\ 0.6317 & 0.0613 & 0.7720 & -0.6317 & 0.0702 & 0.7057 \\ 0.0702 & 0.6745 & 0.6317 & 0.7720 & 0.7720 & 0.0642 \end{bmatrix}$$

The scale factor of each accelerometer

$$s_i = [0.01 \quad 0.02 \quad 0.01 \quad 0.5 \quad 0.06 \quad 0.02]^T$$

The results are nearly the same as the preset parameters.

Then the rigid body is rotated and three sets of data are collected to identify the position error of each accelerometer. The simulated results of the three tests are shown in Table 6-4.

	A_{e1}	A_{e2}	A_{e3}	A_{e4}	A_{e5}	A_{e6}
Test 1	1.1875	7.2464	7.2568	9.4581	8.5258	1.1417
Test 2	8.2181	8.548	2.3324	1.5871	-5.2563	-6.5011
Test 3	6.827	1.7935	9.2929	-4.194	2.0472	7.6242

Table 6-4 Simulated values of the three rotation tests

There are 18 equations related to the 18 variables of the position of the accelerometers. Take $i=1$ as an example,

$$\frac{A_{e1}^{(1)} - b_1}{1+s_1} = \theta_{r1}^T \begin{bmatrix} 0 & 0 & -a_g \end{bmatrix}^T + \theta_{r1}^T \begin{bmatrix} -u_{r1x} & -u_{r1y} & 0 \end{bmatrix}^T$$

$$\frac{A_{e1}^{(2)} - b_1}{1+s_1} = \theta_{r1}^T \begin{bmatrix} -a_g & 0 & \delta \end{bmatrix}^T + \theta_{r1}^T \begin{bmatrix} 0 & -u_{r1y} & -u_{r1z} \end{bmatrix}^T$$

$$\frac{A_{e1}^{(3)} - b_1}{1+s_1} = \theta_{r1}^T \begin{bmatrix} 0 & -a_g & \delta \end{bmatrix}^T + \theta_{r1}^T \begin{bmatrix} -u_{r1x} & 0 & -u_{r1z} \end{bmatrix}^T$$

$$\frac{A_{e1}^{(1)} - A_{e1}^{(2)} + A_{e1}^{(3)} - b_1}{1+s_1} = \theta_{r1}^T \begin{bmatrix} a_g & -a_g & -a_g \end{bmatrix}^T + \theta_{r1}^T \begin{bmatrix} -2u_{r1x} & 0 & 0 \end{bmatrix}^T$$

$$u_{r1x} = \frac{-1}{2\theta_{r1x}} \left(\frac{A_{e1}^{(1)} - A_{e1}^{(2)} + A_{e1}^{(3)} - b_1}{1+s_1} - \theta_{r1}^T \begin{bmatrix} a_g & -a_g & -a_g \end{bmatrix}^T \right)$$

When u_{r1x} is known, u_{r1y} and u_{r1y} can be obtained from

$$u_{r1y} = \frac{-1}{\theta_{r1y}} \left(\frac{A_{e1}^{(1)} - b_1}{1+s_1} - \theta_{r1}^T \begin{bmatrix} 0 & 0 & -a_g \end{bmatrix}^T + \theta_{r1x} u_{r1x} \right)$$

$$u_{r1z} = \frac{-1}{\theta_{r1z}} \left(\frac{A_{e1}^{(3)} - b_1}{1+s_1} - \theta_{r1}^T \begin{bmatrix} 0 & -a_g & \delta \end{bmatrix}^T + \theta_{r1x} u_{r1x} \right)$$

Here are the results

$$u_{ri} = \begin{bmatrix} 0.0054 & 0.0054 & -0.0594 & 0.0594 & 0.0054 & 0.0054 \\ 0.0054 & -0.0594 & 0.0054 & 0.0054 & 0.0486 & 0.0054 \\ -0.0486 & 0.0054 & -0.0054 & 0.0054 & 0.0054 & 0.0594 \end{bmatrix}$$

Once we know the bias error, scale factor error, real position and sensing direction of each accelerometer, compensator is applied to calculate the moving trajectory. The following figures show the performance.

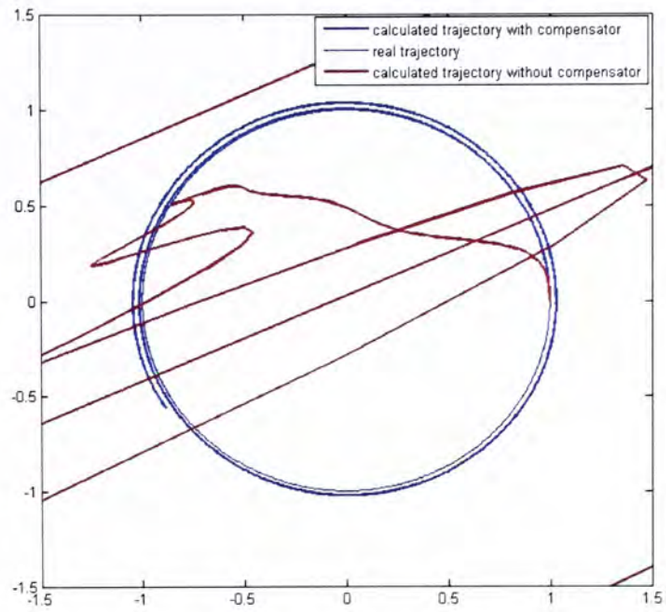


Figure 6-3 Calculated trajectory with compensator

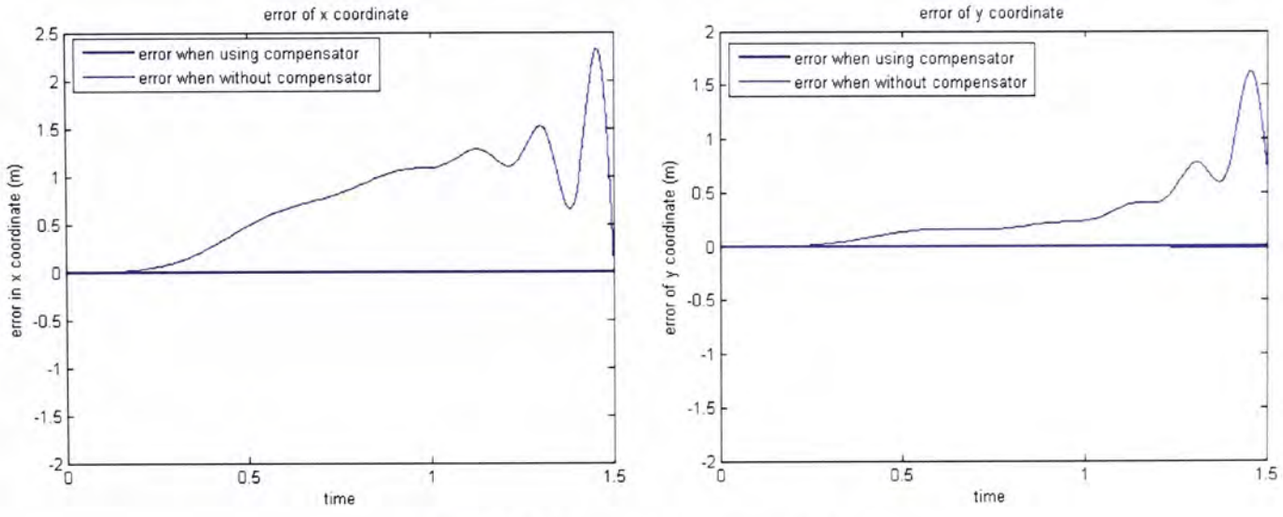


Figure 6-4 Error of values on each axis

The figures show that the compensator is able to reduce the effect of the errors. The resulting trajectory is nearly the same as the real path.

6.2 Fuzzy rule based motion state detector

In calculating the linear and angular displacement in each DOF, at least double integration is needed. The slight error in the acceleration data is magnified and resulted in large error in the displacement. In other words, the accuracy can be enhanced if less integration is needed. The idea of utilizing motion state detector is similar to that presented in section 4.1. It is to add some constraints in the calculating process to achieve utilizing less integration.

6.2.1 Relation of data in different motions

To implement the motion state detector, the characteristics of difference motions should be known. Here we start from the basic algorithm of the GF-INS. Obviously it gives some insights of the relation between the acceleration data and the motion. From the equations (5-23) and (5-24), we have the following six equations,

$$\text{I.}\quad \dot{\omega}_1 = \frac{1}{2\sqrt{2}L}\left(A_1 - A_2 + A_5 - A_6\right) \tag{6-13}$$

$$\text{II.}\quad \dot{\omega}_2 = \frac{1}{2\sqrt{2}L}\left(-A_1 + A_3 - A_4 - A_6\right) \tag{6-14}$$

$$\text{III.}\quad \dot{\omega}_3 = \frac{1}{2\sqrt{2}L}\left(A_2 - A_3 - A_4 + A_5\right) \tag{6-15}$$

$$\text{IV.}\quad P_1 = \frac{1}{2\sqrt{2}}\left(A_1 + A_2 - A_5 - A_6\right) + L\left(\omega_2\omega_3\right) \tag{6-16}$$

$$\text{V.}\quad P_2 = \frac{1}{2\sqrt{2}}\left(A_1 + A_3 - A_4 + A_6\right) + L\left(\omega_1\omega_3\right) \tag{6-17}$$

$$\text{VI.}\quad P_3 = \frac{1}{2\sqrt{2}}\left(A_2 + A_3 - A_4 - A_5\right) + L\left(\omega_1\omega_2\right) \tag{6-18}$$

These six equations are the angular accelerations and linear accelerations with respect to the body frame. Assume the motion of mobile unit is started from rest,

the change of the six equations means that the change of the motion state of the mobile unit. Therefore, observation on the change of value of the equations can give some insight on motion state.

For instance, if the mobile unit rotates about z-axis of the body frame, assuming it starts from rest, equations I, II, IV, V and VI should be zero. However, there are net acceleration readings in those equations, as shown in the experiment in chapter five. To solve this, fuzzy logic is applied. The advantage of fuzzy logic system is that it includes the knowledge from human and allows some degree of approximation, which is useful in determining the state of the motion.

The principle of the detector is introduced as follow. Firstly, the steady state detector described in chapter four is utilized to detect when the object moves. If the object starts moving, check the value of the six equations. The first 16 samples of the motion are utilized. Let the values of the equations are $e_I, e_{II}, e_{III}, e_{IV}, e_V$ and e_{VI}.

Taking the previously stated rotation example, the values of the six equations are as follow.

e_I	e_{II}	e_{III}	e_{IV}	e_V	e_{VI}
2.7303	2.3075	15.0503	-0.2653	-0.1211	-0.3602

Table 6-5 Values of the six equations just after the unit starts to move

Obviously, the value of equation III is dominant. Therefore, it is reasonable to assume the state of the motion is rotation about z-axis of the body frame. The aim of this is to reduce the number of integration. In this example, because of the knowledge about the motion and the starting condition, the integrations involving linear motion along all axes and the angular motion about x and y axes can be neglected. In details, we can set the following values be zero.

$$\dot{\omega}_1 = \dot{\omega}_2 = 0$$

$$\omega_1 = \omega_2 = 0$$

$$P_1 = P_2 = P_3 = 0$$

$$\dot{T}_1 = \dot{T}_2 = \dot{T}_3 = 0$$

To implement this detection process, fuzzy rules based motion state detector is developed. The aim is to detect the state of motion, in order to simplify the algorithm of GF-INS. And finally reduce the error and enhance the accuracy.

6.2.2 Fuzzy system

When the mobile unit starts to move, the values of the six equations are calculated with the first 16 samples. Then those values are the input to a set of fuzzy rules to determine the state of motion. A membership function is designed to map each input to its corresponding membership values. Gaussian membership function is utilized. It has the follow form.

$$\mu_i^l(x_i) = \exp\left(-\left(\frac{x_i - \bar{x}_i^l}{\sigma_i^l}\right)^2\right) \tag{6-19}$$

Where l is the number of rules, i is the number of parameters, x_i is the input, $\bar{x}_i^l$ and σ_i^l are utilized to define the membership function. Other than Gaussian membership function, product inference engine, singleton fuzzifier and center average defuzzifier are utilized in this fuzzy system. The output of the system is,

$$f(x) = \frac{\sum_{l=1}^{M} \bar{y}^l \left[\prod_{i=1}^{n} \exp\left(-\left(\frac{x_i - \bar{x}_i^l}{\sigma_i^l}\right)^2\right)\right]}{\sum_{l=1}^{M} \left[\prod_{i=1}^{n} \exp\left(-\left(\frac{x_i - \bar{x}_i^l}{\sigma_i^l}\right)^2\right)\right]} \tag{6-20}$$

Where $\bar{y}^l$ is utilized to defined the output membership function,

$$\mu^l(y) = \exp\left(-\left(y - \bar{y}^l\right)^2\right) \tag{6-21}$$

There are twelve fuzzy rules are designed for this system.

1. ***IF I*** *is small &* ***II*** *is small &* ***III*** *is large &* ***IV*** *is large &* ***V*** *is large &* ***VI*** *is small* ***THEN*** *the unit is in x-y plan motion,*
2. ***IF I*** *is small &* ***II*** *is small &* ***III*** *is small &* ***IV*** *is large &* ***V*** *is large &* ***VI*** *is small* ***THEN*** *the unit is in x-y plan translation,*
3. ***IF I*** *is small &* ***II*** *is small &* ***III*** *is small &* ***IV*** *is large &* ***V*** *is small &* ***VI*** *is small* ***THEN*** *the unit is in x-axis translation,*
4. ***IF I*** *is small &* ***II*** *is small &* ***III*** *is small &* ***IV*** *is small &* ***V*** *is large &* ***VI*** *is small* ***THEN*** *the unit is in y-axis translation,*
5. ***IF I*** *is small &* ***II*** *is small &* ***III*** *is large &* ***IV*** *is small &* ***V*** *is small &* ***VI*** *is small* ***THEN*** *the unit is in z-axis rotation,*
6. ***IF I*** *is large &* ***II*** *is small &* ***III*** *is small &* ***IV*** *is small &* ***V*** *is large &* ***VI*** *is large* ***THEN*** *the unit is in y-z plan motion,*
7. ***IF I*** *is small &* ***II*** *is small &* ***III*** *is small &* ***IV*** *is small &* ***V*** *is large &* ***VI*** *is large* ***THEN*** *the unit is in y-z plan translation,*
8. ***IF I*** *is small &* ***II*** *is small &* ***III*** *is small &* ***IV*** *is small &* ***V*** *is small &* ***VI*** *is large* ***THEN*** *the unit is in z-axis translation,*
9. ***IF I*** *is large &* ***II*** *is small &* ***III*** *is small &* ***IV*** *is small &* ***V*** *is small &* ***VI*** *is small* ***THEN*** *the unit is in x-axis rotation,*
10. ***IF I*** *is small &* ***II*** *is large &* ***III*** *is small &* ***IV*** *is large &* ***V*** *is small &* ***VI*** *is large* ***THEN*** *the unit is in x-z plan motion,*
11. ***IF I*** *is small &* ***II*** *is small &* ***III*** *is small &* ***IV*** *is large &* ***V*** *is small &* ***VI*** *is large* ***THEN*** *the unit is in x-z plan translation,*
12. ***IF I*** *is small &* ***II*** *is large &* ***III*** *is small &* ***IV*** *is small &* ***V*** *is small &* ***VI*** *is small* ***THEN*** *the unit is in y-axis rotation,*

These twelve rules define the relation between the values of the six equations and different motions. However, they are based on the assumption that the motion is started from stationary. The calibrations on the accelerations and velocities on the non-moving axes are not workable unless the assumption holds. This is the limitation of the detector.

On the other hand, the performance of this fuzzy rule system depends on the definitions of the membership functions. A better way to define these functions is utilizing experimental data. Therefore, a training method is needed for obtaining appropriate membership functions.

6.2.3 Membership function training with gradient descent

As stated in equations (6-19) and (6-21), there are three parameters $\bar{x}_i^l$, σ_i^l and $\bar{y}^l$ which defined the membership functions of the fuzzy system. The values of these parameters can be simply assigned by expert. The other way is utilizing input-output pairs $\left(x_o^P; y_o^P\right)$ to train the membership function. The training method proposed here is utilizing gradient descent.

Gradient descent is an iterative algorithm that seeks to decrease the value of the objective function in each iteration. It relies on the fact that from any point the objective function decreases most rapidly in the direction of the negative gradient vector of its parameters at that point. If we have objective function $E(z)$, where parameters are $z = (z_1, z_2, \ldots, z_p)$, then this vector is

$$\left(\frac{-\partial E}{\partial z_1}, \frac{-\partial E}{\partial z_2}, \ldots, \frac{-\partial E}{\partial z_p}\right) \tag{6-22}$$

If $z_i(t)$ is the value of the i-th parameter at iteration t, then the algorithm decrease the objective function by

$$z_i(t+1) = z_i(t) - K\frac{\partial E(z)}{\partial z_i} \qquad i=1, \quad ,p \tag{6-23}$$

Where K is a constant which controls how much the parameters are altered at each iteration. As the iterations proceed, the objective function converges to a local minimum.

Implementing gradient descent in fuzzy membership function training is firstly proposed by Nomura et al [53]. Another similar work can also found in [54].

There are several steps for training with gradient descent.

1. Initialize parameters

Specify the initial parameters $\bar{y}^l(0)$, $\bar{x}_i^l(0)$ and $\sigma_i^l(0)$. These initial parameters may be determined according to the expert.

2. Calculate the output of the fuzzy system with input output pairs

For a given input-output pair $\left(x_0^p; y_0^p\right)$, $p = 1,2,\ldots$, and at the q'th stage of training, $q = 0,1,2,\ldots$, input x_0^p to the fuzzy system and compute the outputs. That is, compute

$$z^l = \prod_{i=1}^{n} \exp\left(-\left(\frac{x_0^p - \bar{x}_i^l(q)}{\sigma_i^l(q)}\right)^2\right) \tag{6-24}$$

$$b = \sum_{l=1}^{M} z^l \tag{6-25}$$

$$a = \sum_{l=1}^{M} \bar{y}^l(q) z^l \tag{6-26}$$

$$f = a/b \tag{6-27}$$

3. Update the parameters

Use the following equations to update the parameters.

$$\bar{y}^l(q+1) = \bar{y}^l(q) - \alpha \frac{f-y}{b} z^l \tag{6-28}$$

Where $l = 1,2,\ldots M$ and $q = 0,1,2,\ldots$

$$\bar{x}_i^l(q+1) = \bar{x}_i^l(q) - \alpha \frac{f-y}{b}\left(\bar{y}^l(q) - f\right) z^l \frac{2\left(x_{0i}^p - \bar{x}_i^l(q)\right)}{\left(\sigma_i^l(q)\right)^2} \tag{6-29}$$

Where $i = 1,2,\ldots n$, $l = 1,2,\ldots M$ and $q = 0,1,2,\ldots$

$$\sigma_i^l(q+1) = \sigma_i^l(q) - \alpha \frac{f-y}{b}\left(\bar{y}^l(q) - f\right) z^l \frac{2\left(x_{0i}^p - \bar{x}_i^l(q)\right)^2}{\left(\sigma_i^l(q)\right)^3} \tag{6-30}$$

Where $i = 1,2,\ldots n$, $l = 1,2,\ldots M$ and $q = 0,1,2,\ldots$

4. Calculate the error function

Repeat by going to Step 2 with $q = q+1$, until the error $\left|f - y_0^p\right|$ is less than a preset number ε, or until the q equals a preset number.

5. Repeat the process with another input-output pair

Repeat by going to Step 2 with $p = p+1$, that is, update the parameters using the next input-output pair $\left(x_0^{p+1}; y_0^{p+1}\right)$.

To training the membership functions, at least one input-output pair for each rule are utilized to enhance the accuracy the detector.

6.3 Experimental results and discussion

6.3.1 Configuration errors

In order to identify the scale factor error, bias error, sensing direction error and position error, the GF-INS sensor box is tested with the robot arm platform. There are nine tests including six stationary tests and three rotation tests which are listed in section 6.1. First of all, the six stationary tests are undergone and the results are listed in Table 6-6.

	A_{e1}	A_{e2}	A_{e3}	A_{e4}	A_{e5}	A_{e6}
Test 1	-0.05	0.2913	6.6513	-6.4547	7.2339	-7.1552
Test 2	6.6481	-6.4707	7.3082	-7.0267	0.3300	0.4715
Test 3	6.7653	-6.9644	-0.4738	-0.2369	6.7653	-6.9644
Test 4	6.5075	-6.8031	0.1833	-0.6185	-7.2449	7.1491
Test 5	6.7825	-6.8105	-6.9921	6.9658	-0.5448	0.4583
Test 6	0.0966	0.1449	-7.2211	7.3915	6.4253	-6.3429

Table 6-6 Experimental results for the six stationary tests

Then the calculated scale factor error, bias error and sensing direction are,

$$s_i = \begin{bmatrix} -0.0075 & -0.0093 & -0.0100 & 0.0004 & -0.0054 & -0.0107 \end{bmatrix}^T$$

$$b_i = \begin{bmatrix} 0.0861 & 0.2101 & -0.1848 & -0.1378 & -0.0235 & 0.0824 \end{bmatrix}^T$$

$$\theta_{ri} = \begin{bmatrix} 0.6733 & 0.7377 & -0.0122 & 0.0409 & -0.7155 & -0.7530 \\ 0.7392 & -0.0073 & 0.7071 & -0.7336 & -0.0514 & 0.6580 \\ -0.0175 & 0.6751 & 0.7071 & 0.6784 & 0.6968 & -0.0025 \end{bmatrix}$$

To find the actual position of the accelerometers, the three rotation tests are performed. The target data is the accelerations when the box is in constant angular velocity. Since there is vibration on the acceleration data when the object is moving, as shown in section 4.1.1, it is difficult to obtain the exact value. Here,

the mean value over the period with constant angular velocity is utilized. The results are shown below.

	A_{e1}	A_{e2}	A_{e3}	A_{e4}	A_{e5}	A_{e6}
Test 1	-0.08484	6.7696	6.6602	6.5179	6.788	0.0582
Test 2	6.6221	8.0467	0.3950	0.9936	-6.3096	-7.226
Test 3	7.2615	0.8060	7.3772	-6.6586	0.1775	6.4648

Table 6-7 Experimental data for the three rotational tests

Then the calculated positions of the accelerometers are,

$$u_{ri} = \begin{bmatrix} 0.0018 & 0.0016 & -0.8666 & 0.6597 & 0.0173 & 0.0002 \\ -0.0013 & 0.1825 & 0.0134 & 0.0372 & 0.0515 & 0.0003 \\ -0.1904 & -0.1585 & -0.1709 & -0.1961 & -0.1559 & 0.0820 \end{bmatrix}$$

However, the calculated values are not reasonable in some cases. It is because the size of the sensor box is around $10cm \times 10cm \times 10cm$. The reason is the difficulty of getting the accurate values of the accelerations when the object is moving with constant velocity. Small variant on the measurement causes inaccuracy on the calculated result. To solve this, manual measurement is performed to find the actual position of the accelerometers. Here are the results.

$$u_{ri} = \begin{bmatrix} 0.002 & -0.001 & -0.0529 & 0.05283 & 0.003 & 0 \\ -0.001 & -0.0521 & -0.005 & 0.002 & 0.05254 & 0.001 \\ -0.052615 & 0 & 0.002 & 0 & 0.004 & 0.052615 \end{bmatrix}$$

6.3.2 Compensator

To test the performance of the compensator, the scale factor error, bias error, sensing direction error and position error found in previous section are utilized here to form the compensator. The test performed in section 5.3 is done again to compare the effect of the compensator. The compensated formula is discussed in section 6.1.3. It is,

$$A_i = \left(J_r J^{-1}\right)^{-1}\left[\frac{A_{ei}-b_i}{1+s_i} + J_r J^{-1}\begin{bmatrix}\theta_1{}^T\Omega^2 u_1\\ \vdots \\ \theta_6{}^T\Omega^2 u_6\end{bmatrix} - \begin{bmatrix}\theta_{r1}{}^T\Omega^2 u_{r1}\\ \vdots \\ \theta_{r6}{}^T\Omega^2 u_{r6}\end{bmatrix}\right]$$

The ideal size ($2L$) of the cube is set as L=5cm. The ideal sensing directions and positions are the same as the configuration. Then the compensated acceleration is shown as follow.

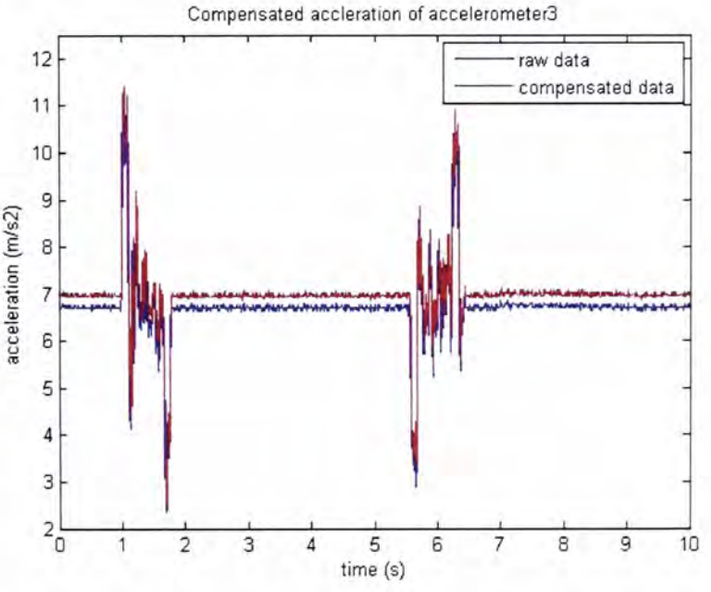


Figure 6-5 Compensated acceleration of accelerometer3

The above shows one of the readings of the accelerometers is successfully compensated. The resulting calculated displacement is shown in Figure 6-6. The results show that both the results in x and y axes are improved. This experiment illustrates that the implementation of compensator can reduce the error which is due to the scale factor error, bias error and configuration errors.

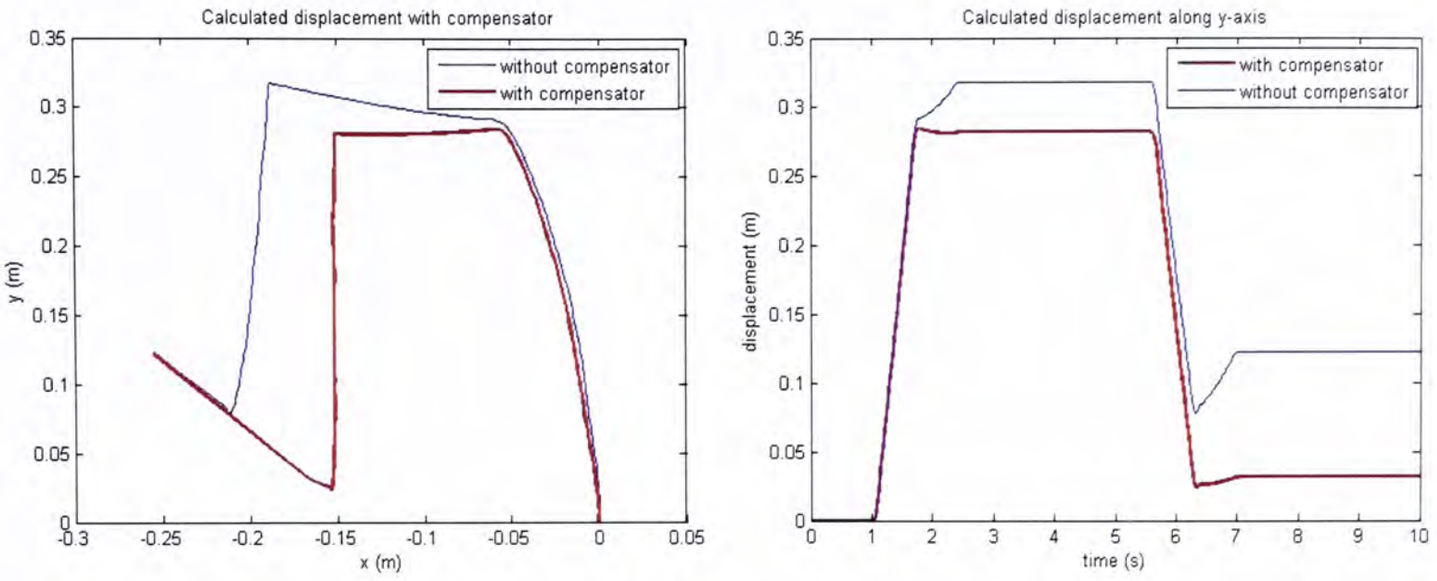
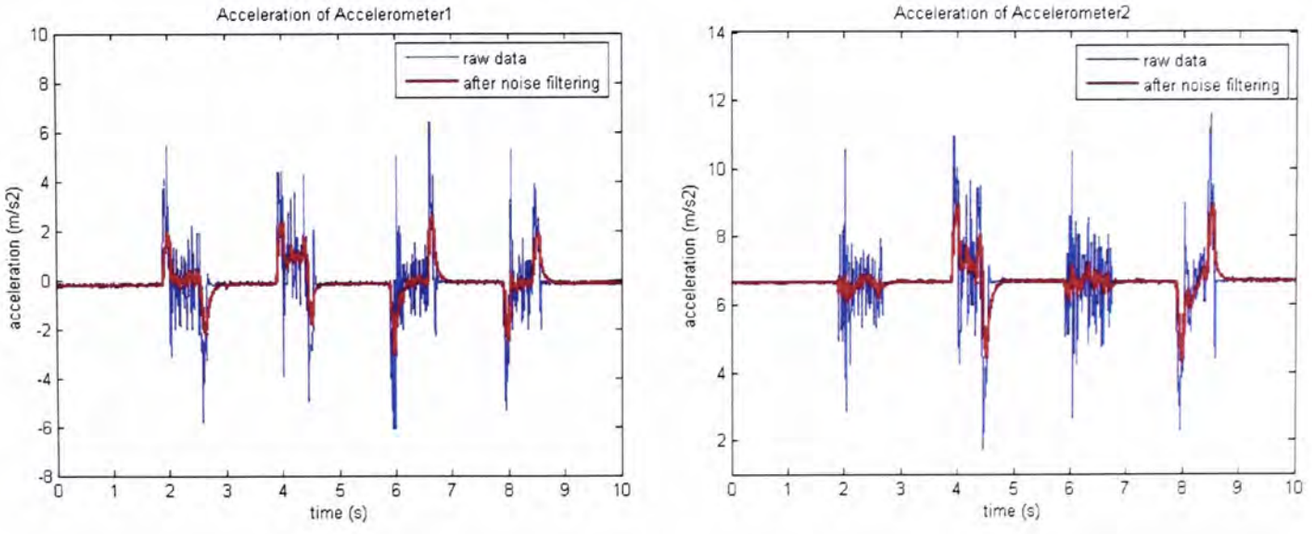


Figure 6-6 Calculated displacement

6.3.3 Fuzzy rule based motion state detector

A translation in x-y plan is performed. The unit moves in a return rectangle shape in the x-y plan with dimensions 0.257m in y and 0.206m in x. The four motions are well separated. The sensed accelerations are shown below.



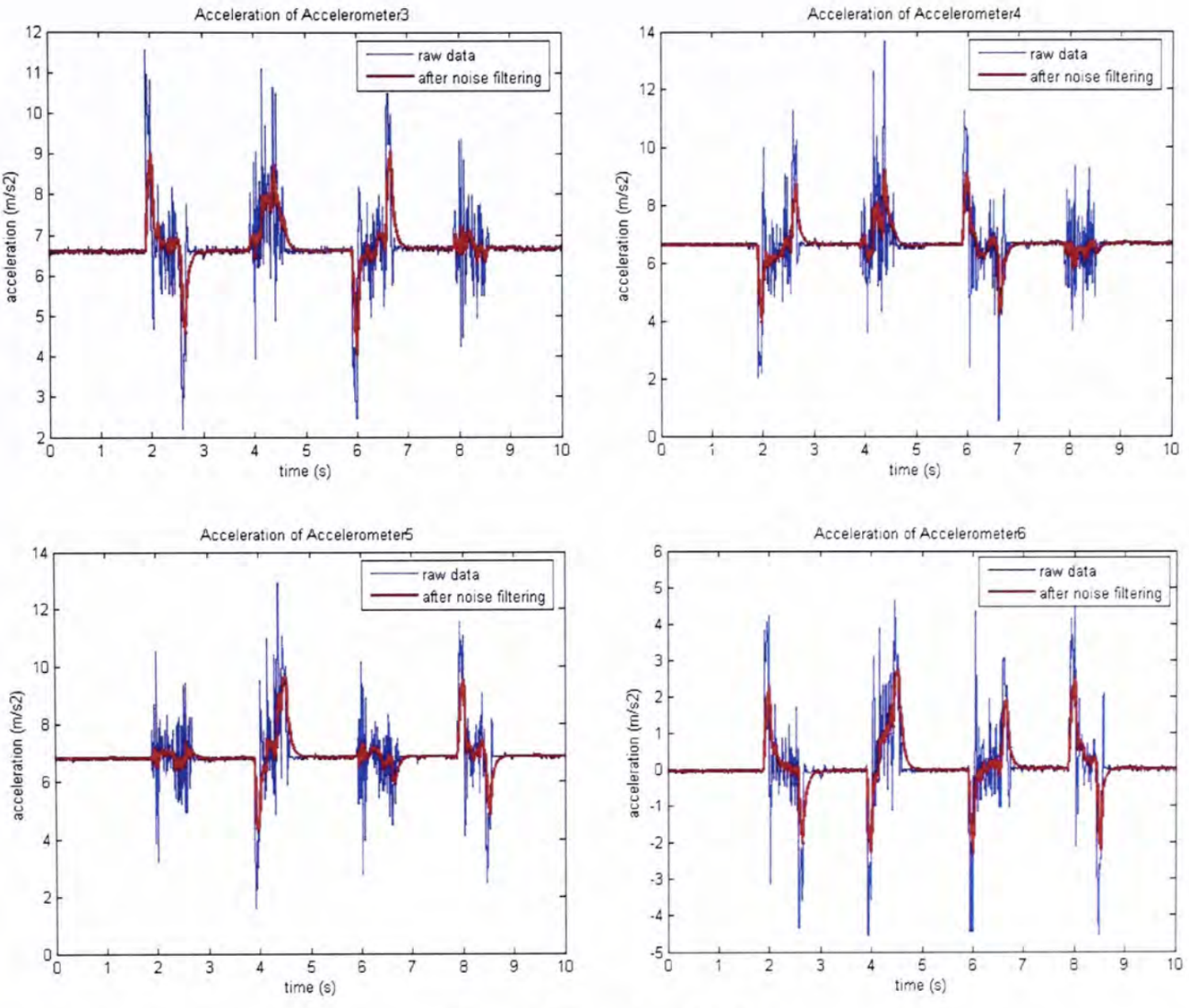


Figure 6-7 Sensed accelerations

Firstly, the data are processed with the steady state detector and the Kalman filter. Then the values of the equations (6-13) to (6-18) are obtained when the motion starts. The corresponding values are listed below.

motion	e_I	e_{II}	e_{III}	e_{IV}	e_V	e_{VI}
1	-0.91652	2.4971	2.9509	-0.25528	**4.0261**	-0.22264
2	-1.0314	-2.5428	-2.9874	**3.9932**	0.28329	-0.10755
3	-0.87727	-0.36769	-1.1352	-0.18732	**-4.4765**	-0.12106
4	1.9479	1.1755	2.1643	**-4.3835**	-0.04142	0.077458

Table 6-8 Normalized values of the six characteristic equations

One may note that the values of the first three equations regarding angular motions are usually greater than the later three values related to linear motions. It is due to the linear combination of inaccurate acceleration data. To avoid inappropriate detection on the state of motion, the fuzzy rule based system and gradient descent training method described previously are utilized. The system is trained with real experimental data to enhance the performance of detection.

In this case, the motions detected are firstly y-axis translation, then x-axis translation, and then y-axis translation again and finally is x-axis translation. Therefore, the corresponding constraints are added in the integration process. The calculated results are as follow.

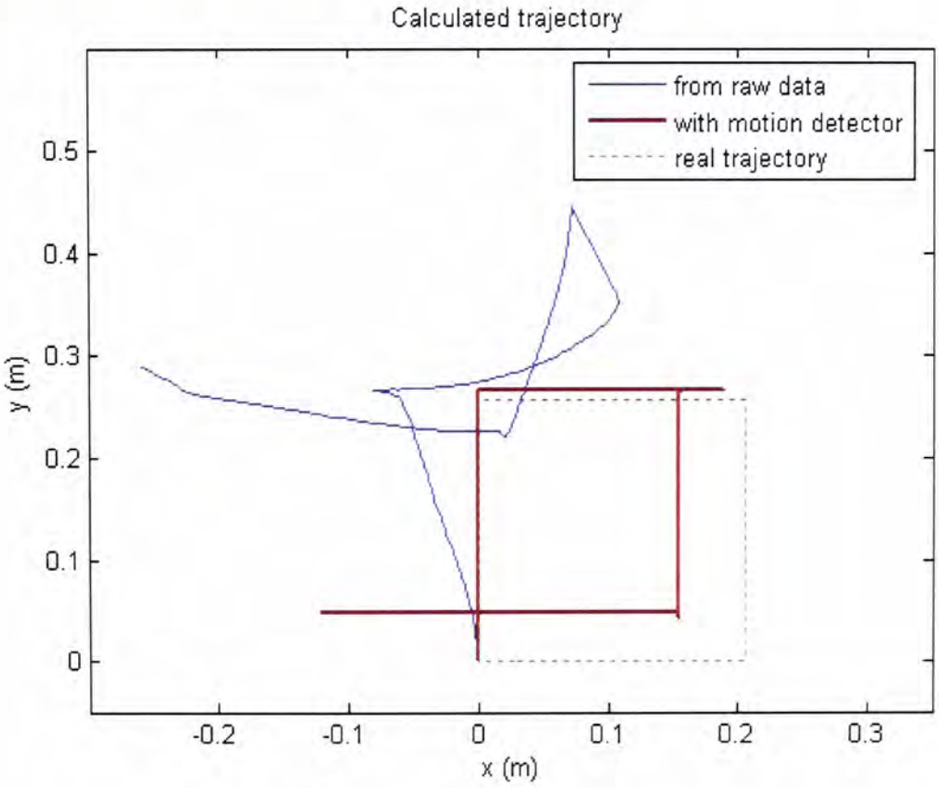


Figure 6-8 Calculated displacement of the mobile unit

The result is in rectangular shape that matched the real path. However, the detected displacement is slightly different from the real one, 0.257m x 0.206m. But it is much better than the result without the motion detector.

6.3.4 Comparing the performance of both methods

To illustrate the individual performance of the configuration compensator and the motion state detector, two experiments are shown in the previous sections. Here, the two performances are compared and the results combining both methods are shown.

The experimental data described in section 6.3.3 are utilized. The robot arm performs a rectangular trajectory on the x-y plan with the distances 0.257m in y and 0.206m in x.

The data are firstly processed by the steady state detector and the Kalman filter. The trajectory is then calculated based on four schemes; the basic algorithm, configuration error compensation, motion detection and finally with combination of both methods. The results are shown below.

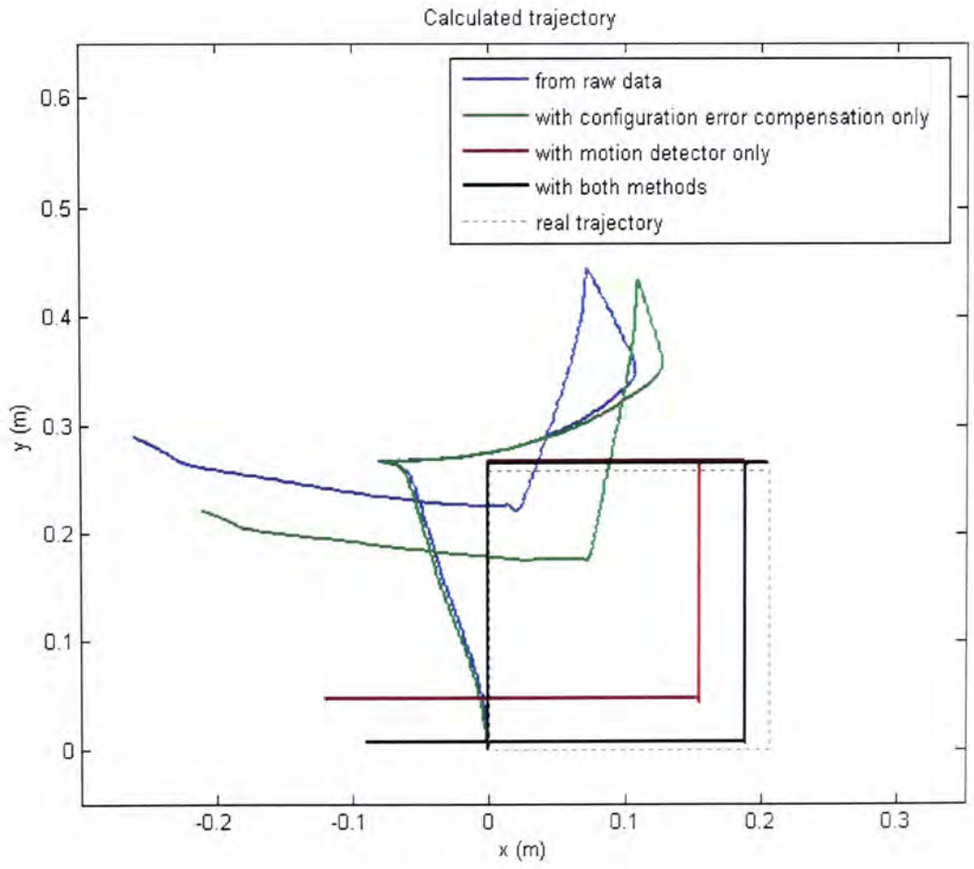


Figure 6-9 Calculated trajectory with different error reduction methods

The corresponding x and y displacement are shown below.

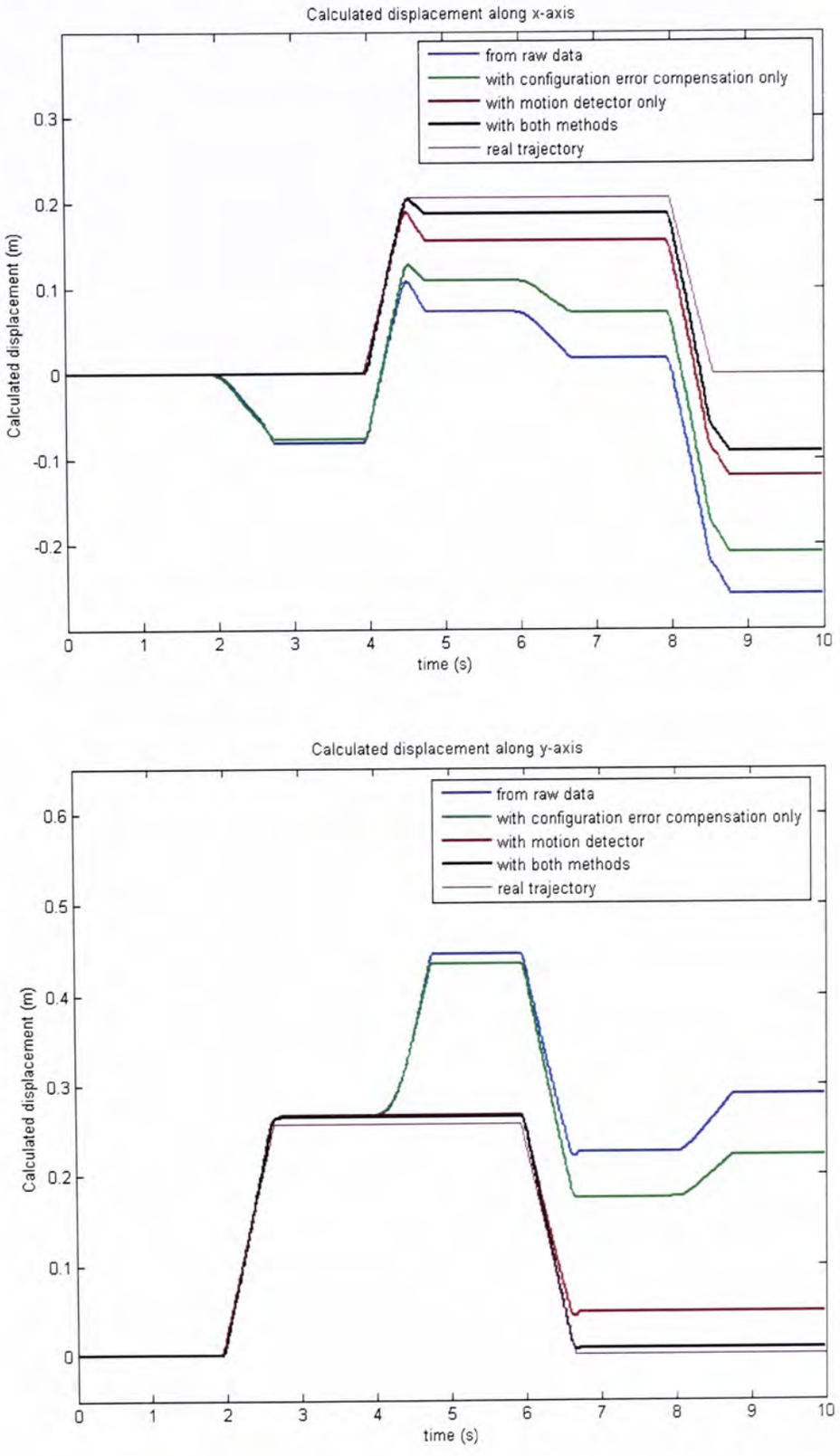


Figure 6-10 Calculated displacement on x and y axis

From the results, we can conclude that the error of utilizing the basic algorithm is the largest. It is because the setup is slightly different from the ideal configuration. Therefore the error is reduced when the configuration error is compensated. However, the result is still not satisfied in position determination. The implementation of motion state detector adds some constraints so that integrations on some inaccurate acceleration data are ignored. The trajectory detected is greatly improved. Besides, the combination of both configuration error reduction and motion detector does give the best performance compared with their individual results.

6.3.5 Comparing GF-INS and one dimensional INS

In order to compare the performance of the one dimensional INS developed in chapter four and GF-INS, both setups are tested in a same motion. The noise reduction methods proposed in chapter five and this chapter are all implemented to give the best results. The results are shown in Figure 6-11 and the details are shown in Table 6-9.

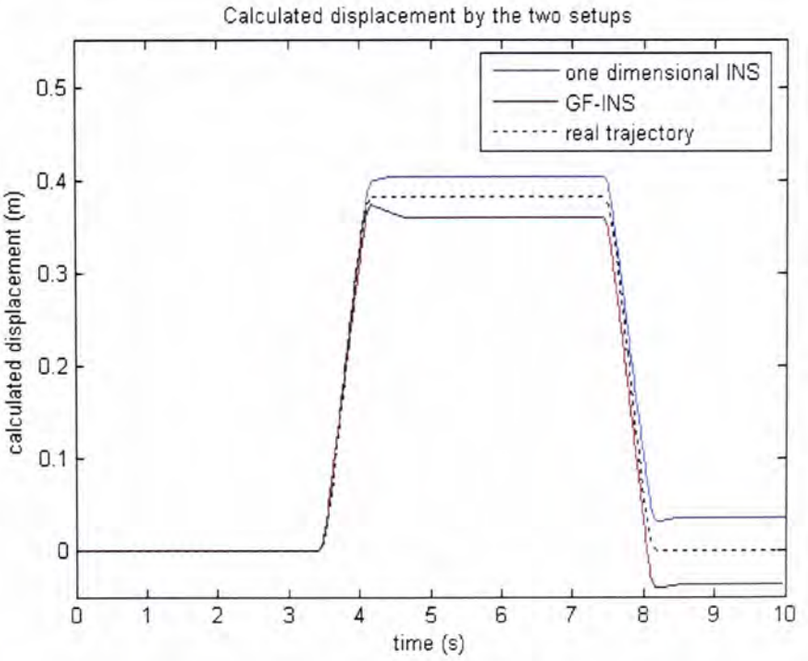


Figure 6-11 Calculated displacement by the two setups

	Real position	One dimensional INS	GF-INS
After 2s	0	0	0
	Error	**0**	**0**
After 5s	0.388	0.4035	0.3599
	Error	**0.0155**	**0.0281**
After 10s	0	0.0346	-0.0377
	Error	**0.0346**	**0.0377**

Table 6-9 Comparison of the performance of the two setups

Both setups give acceptable results. The result of one dimensional INS is better than that of GF-INS. Actually, implementing the motion state detector on GF-INS reduce its dimension from six to one in this case. Because of the linear combination of the accelerations in the algorithm of GF-INS, slight inaccurate sensing in one of the six accelerometers would affect the detected result. That is the reason that GF-INS is less stable in position determination compared with the one dimensional INS. However, the advantage of GF-INS is that it can sense the six dimensional movements without adding another type senor such as gyroscopes.

6.3.6 Discussion

These experiments illustrate the work discussed in this chapter. The first one shows a way to identify the configuration error as well as the scale factor error and the bias error. It is successful in finding sensing direction error. However, difficulties are found when identifying the position error. It is because it is hard to obtain a stable value of acceleration when the unit is in constant angular velocity. The sensitivity of the accelerometers is found very high. The signal is vibrating once the unit is moving. This actually affects the accuracy of the system. To identify the real position of the accelerometers, manual measurement with digital calipers is performed.

The second experiment illustrates the performance of the compensator. This compensator design is modified from the original one which proposed by Tan and Pak [45]. The effect of scale factor error and bias error are added to enhance the performance. From the results of the experiment, the compensator performs well and the calculated displacement is more accurate.

The third one demonstrates the usage of motion state detector. The values of the six characteristic equations just after the motion started are the input of the fuzzy rules based system. The results show that the performance can be improved by the implementation of the state detector. Although there are still errors compared to the real path, it is highly improved from result of not using. One challenge of the state detector is the design of the membership functions. Other than assigned by expert, the method of utilizing sample data and gradient decent is introduced.

However, there are drawbacks of the motion state detector. Firstly, it can only detect the motion starting from steady state. It is because the characteristic equations only represent the linear and angular accelerations. If the motion is changing without separation, it is hard to add the constraints on the velocities as the previous velocities are not zero. Another drawback is that it takes 16 sampling times to detect the motion state. That means there is time delay of getting the position results. It is not a matter for off-line system but is important for on-line system.

The forth one shows the performance of combining the configuration error compensation and the motion state detector. It is found that GF-INS gives the best performance when both methods are implemented.

The last experiment provides a comparison of the GF-INS and the one dimensional INS. It is reasonable that the one dimensional INS is more accurate than the GF-INS. But GF-INS still has its own advantage that able to sense six dimensional motions without adding other kind of sensors.

6.4 Summary

The aim of this chapter is to propose two methods to enhance the performance of the GF-INS in position determination.

The first method is the configuration error compensation. It starts from identifying the scale factor error, bias error, sensing direction error and the position error of the six accelerometers. Six stationary tests with difference orientations and three rotation tests along different axes are performed. An improved compensator to compensate those errors is developed. To show the feasibility of the compensator, simulation is performed. The results show that it works well to reduce the errors of calculated displacement.

The second method is a fuzzy rule based motion state detector. It is the extension of the fuzzy logic based steady state detector described in chapter four. It makes use of six characteristic equations and combines to the steady state detector. Once the unit starts moving, the values of the six equations are inputted to a fuzzy rule based system. A method of gradient descent to train the membership functions with sample input-output pairs is described.

Then five experiments are carried out to test the compensator and the motion detector. The results are shown that both compensator and motion detector are workable in improving the performance of the GF-INS in position determination.

Chapter 7

Conclusions and Future works

The objective of this thesis is to build a system based on INS to determine the distance traveled by mobile unit. The principle of INS and GF-INS are studied and an INS and a GF-INS are constructed with MEMS accelerometer. The hardware should be low cost and with acceptable accuracy.

The work is started by investigating and evaluating the performance of the MEMS accelerometer in position determination. To do so, a one dimensional INS is developed. The accelerometer utilized is ADXL103 from Analog Devices Inc.. One the other hand, the integrated device MICA Mote is utilized to ease the hardware implementation. It contains a microprocessor, eight channels ADC, flash memory and radio module. The methods of initial calibration, high frequency noise filtering and configuration error compensation are discussed. The device is tested with a robot platform which can move in five DOF and is controlled by user. The results illustrate that directly double integrated from raw data gives the worst result. The low pass filter has little effect on the calculated position. But the configuration error compensation is found an effective way to reduce the error. The error is reduced more than ten times after ten seconds. This gives the first insight on the performance of the MEMS accelerometer.

Since the results of the one dimensional INS is not good enough, further methods are proposed and investigated. It is noted that there are two kinds of error sources. One is the slightly vibration on the acceleration data when the object is not moving. Because of the double integrations, the error enlarges and the accuracy is affected. To solve this, a fuzzy logic based steady state detector is proposed. By determining the standard deviation of the acceleration data, the steady state and the moving state are classified. With the knowledge of not moving, constraints can be added to the acceleration and velocity data to reduce the effect of the signal vibration when the unit is not moving. The second error source is the random

noise of the accelerometer itself. A Kalman filter is implemented to reduce this white Gaussian noise. It combines with the steady state detector to remove the noise on both steady state and moving state. The experimental results show that the calculated displacements with these noise reduction methods are very close to the actual paths. Therefore, with suitable signal processing methods, the MEMS accelerometer is acceptable in distance measurement in a short period of time.

As the MEMS accelerometer is suitable in position determination, the one dimensional INS is extended to GF-INS. Utilizing linear accelerometers to determine both linear and angular motion of rigid body has been discussed for forty years. The algorithm of GF-INS is based on a particular cube-shape configuration. It is studied with simulations. Furthermore, hardware of GF-INS is constructed. Six accelerometers are placed in the cube shape as stated in the algorithm. The hardware is tested with the robot arm. It is found that the GF-INS is sensitive to noise. It is because the algorithm depends on the linear combination of the six accelerations. But the actual data is not perfect. Therefore, extra noise reduction methods are needed to enhance the performance.

Two methods for noise reduction are proposed. One is the configuration error compensation. It is a modification from the original one. The method of identifying the actual position is modified for the practicality. The effect of the scale factor error and bias error are added to the compensator to improve the performance. The second one is a fuzzy rule based motion state detector. It is the extension of the steady state detector described before. The values of six characteristic equations are inputted to the fuzzy rule based system. A method utilizing gradient descent is proposed to train the fuzzy membership functions with sample input-output pairs. The training of the membership function aims to increase the accuracy of the detection. Experiments are carried out to illustrate the performance. One is identifying the configuration errors of the hardware. It is found that the modified methods are workable but the accuracy is hard to control particularly in rotation tests. The second test shows that the performance of the

compensator is acceptable and the error of calculated displacement is reduced. The third test shows the principle of the motion state detector. The result is similar to actual path although errors still exists. The forth one shows that combining both configuration error compensation and motion state detector gives the best result on position determination. The comparison on one dimensional INS and GF-INS is given in the last test. It is reasonable that the one dimensional INS is slightly better than the GF-INS in one dimensional motion detection.

To conclude, the principle of INS and GF-INS are studied in this thesis. The performance of MEMS accelerometer in position determination is found acceptable when appropriate signal processing methods are applied. Besides, hardware implementations of both INS and GF-INS are performed. For the noise reduction, fuzzy logic is proposed to form two successful detectors to determine the motion state of the unit. Then Kalman filter is also implemented to reduce the random noise. And the configuration error compensation methods are modified to improve the performance. All the results are verified by experimental data. The resulting setup is able to determine the position of mobile unit in short duration.

For further development, the GF-INS should be tested in different mobile platform. Other noise reduction methods should be studied and implemented. For example, the GF-INS can be periodically recalibrated by external position source such as GPS in outdoor, vision system in indoor environment. The integrated navigation system will be the trend in the near future. The cooperation between INS or GF-INS and other positioning technologies are good direction to enhance the performance of the INS and GF-INS.

Furthermore, the application of GF-INS should be considered. One possible idea is utilizing in indoor human monitoring. It is capable to combine with pedometer and hidden Markov model (HMM) to determine the movement and motion of human. Then with some knowledge of the map of the environment, an integrated system for indoor human monitoring can be developed.

Reference

[1] Wikipedia, "Global Positioning System – Wikipedia, the free encyclopedia," http://en.wikipedia.org/wiki/Global_Positioning_System.

[2] U.S. Coast Guard Navigation Center, "Global Positioning System," http://www.gps.gov.

[3] J. Farrell, M. Barth, "The Global Positioning System and Inertial Navigation," MaGraw-Hill, 1999.

[4] Europa official website, "Energy and Transport – GALILEO," http://ec.europa.eu/dgs/energy_transport/galileo/index_en.htm.

[5] M. S. Grewal, L. R. Weill, A. P. Andrews, "Global Positioning Systems, Inertial Navigation, and Integration," John Wiley & Sons Inc., 2001.

[6] M. Vossiek, L. Wiebking, P. Gulden, J. Wieghardt, C. Hoffmann, "Wireless Local Positioning – Concepts, Solutions, Applications," *in Proc. IEEE Radio and Wireless Conf.*, pp. 219-224, 2003.

[7] A. Stelzer, K. Pourvoyeur, A. Fischer, "Concept and Application of LPM – A Novel 3-D Local Position Measurement System," *IEEE Trans. on Microwave Theory and Techniques*, Vol. 52, Issue 12, pp. 2664-2669, 2004.

[8] H. Saarnisaari, "Some Design Aspects of Mobile Local Positioning Systems," *IEEE Position Location and Navigation Symp.*, pp. 300-309, 2004.

[9] E. Aitenbichler, M. Muhlhauser, "An IR Local Positioning System for Smart Items and Devices," *in Proc. of the 23rd Int. Conf. on Distributed Computing Systems Workshops*, pp. 334-339, 2003.

[10] J. M. Villadangos, J. Urea, M. Mazo, A. Hernandez, C. D. Marziani, M. C. Perez, F. Alvarez, J. J. Garcia, A. Jimenez, I. Gude, "Ultrasonic Local

Positioning System with Large Covered Area," *IEEE Int. Symp. on Intelligent Signal Processing*, pp. 1-6, 2007.

[11] P. Bahl, V. N. Padmanabhan, "RADAR: An In-Building RF-based User Location and Tracking System," *in Proc. Nineteenth Annual Joint Conf. of the IEEE Computer and Communications Societies*, Vol. 2, pp. 775-784, 2000.

[12] A. D. Koutsou, F. Seco, A. R. Jimenez, J. O. Roa, J. L. Ealo, C. Prieto, J. Guevara, "Preliminary Localization Results With An RFID Based Indoor Guiding System," *IEEE Int. Symp. on Intelligent Signal Processing*, pp. 1-6, 2007.

[13] A. Kotanen, M. Hannikainen, H. Leppakoski, T. D. Hamalainen, "Positioning with IEEE 802.11b Wireless LAN," *in 14th IEEE Proc. on Personal, Indoor and Mobile Radio Communications*, Vol. 3, pp. 2218-2222, 2003.

[14] L. Wiebking, M. Glanzer, D. Mastela, M. Christmann, M. Vossiek, "Remote Local Positioning Radar," *IEEE Radio and Wireless Conf.* pp.191-194, 2004.

[15] H. Reddy, M. G. Chandra, P. Balamuralidhar, S. G. Harihara, K. Bhattacharya, E. Joseph, "An Improved Time-of-Arrival Estimation for WLAN-Based Local Positioning," *2nd Int. Conf. on Communication System Software and Middleware*, pp. 1-5, 2007.

[16] M. Marcu, S. Fuicu, A. Girban, "Local Wireless Positioning System," *4th Int. Symp. on Applied Computational Intelligence and Informatics*, pp. 171-176, 2007.

[17] S. Kawakubo, A. Chansavang, S. Tanaka, T. Iwasaki, K. Sasaki, T. Hirota, H. Hosaka, H. Ando, "Wireless Network System for Indoor Human Positioning," *1st Int. Symp. on Wireless Pervasive Computing*, 6 pp, 2006.

[18] J. G. Castano, M. Svensson, M. Ekstrom, "Local Positioning for Wireless Sensors Based on Bluetooth," *IEEE Radio and Wireless Conf.*, pp. 195-198, 2004.

[19] A. Kotanen, M. Hannikainen, H. Leppakoski, T. D. Hamalainen, "Experiments on Local Positioning with Bluetooth," *in Proc. of Int. Conf. on Information Technology: Coding and Computing [Computers and Communications]*, pp. 297-303, 2003.

[20] C. Randell, C. Djiallis, H. Muller, "Personal Position Measurement Using Dead Reckoning," *in Proc. of the Seventh IEEE Int. Symp. on Wearable Computers*, pp. 166-173, 2003.

[21] D. Gebre-Egziahber, "Design and Performance Analysis of a Low-Cost Aided Dead Reckoning Navigator," *Ph.D. Dissertation, Stanford University*, 2004.

[22] A. D. King, "Inertial Navigation – Forty Years of Evolution," *General Electric Company Review*, Vol. 13, No. 3, pp. 140-149, 1998.

[23] K. J. Walchko, "Low Cost Inertial Navigation: Learning to Integrate Noise and Find Your Way," *M.Sc. Thesis, University of Florida*, 2002.

[24] H. Qi, J. B. Moore, "Direct Kalman filtering approach for GPS/INS integration," *IEEE Trans. on Aerospace and Electronic Systems*, Vol. 38, Issue 2, pp. 687-693, 2002.

[25] C. Hide, T. Moore, M. Smith, "Adaptive Kalman filtering algorithms for integrating GPS and low cost INS," *Position Location and Navigation Symp.*, pp.227-233, 2004.

[26] B. Wang, J. Wang, J. Wu, B. Cai, "Study on adaptive GPS/INS integrated navigation system," *Proc. of 2003 IEEE Intelligent Transportation Systems*, Vol. 2, pp. 1016-1021, 2003.

[27] A. Noureldin, R. Sharaf, A. Osman, N. El-Sheimy, "INS/GPS Data Fusion Technique Utilizing Radial Basis Functions Neural Networks," *Position Location and Navigation Symp.*, pp. 280-284, 2004.

[28] A. Hiliuta, R. Jr. Landry, F. Gagnon, "Fuzzy Corrections in a GPS/INS Hybrid Navigation System," *IEEE Trans. on Aerospace and Electronic Systems*, Vol. 40, Issue 2, pp. 591-600, 2004.

[29] T. C. Li, "Analysis of Inertial Navigation System Errors from Van Testing Using an Optimal Kalman Filter/Smoother," *Position Location and Navigation Symp.*, pp. 120-128, 2004.

[30] F. Gul, J. Fang, "Correction Technique for Velocity and Position Error of Inertial Navigation System by Celestial Observations," *Proc. of the IEEE Symp. on Emerging Technologies*, pp. 7-12, 2005.

[31] X. Ma, S. Sukkarieh, J. H. Kim, "Vehicle Model Aided Inertial Navigation," *Proc. of the IEEE Intelligent Transportation Systems*, Vol. 2, pp. 1004-1009, 2003.

[32] M. M. Veth, J. Raquet, "Fusion of Low-Cost Imaging and Inertial Sensors for Navigation," *Proc. of ION GPS*, pp. 1093-1103, 2006.

[33] M. Kourogi, T. Kurata, "Personal Positioning based on Walking Locomotion Analysis with Self-Contained Sensors and a Wearable Camera," *Proc. the second IEEE and ACM Int. Symp. on Mixed and Augmented Reality*, pp. 103-112, 2003.

[34] Y. Yang, J. A. Farrell, "Magnetometer and Differential Carrier Phase GPS-Aided INS for Advanced Vehicle Control," *IEEE Trans. on Robotics and Automation*, Vol. 19, Issue 2, pp. 269-282, 2003.

[35] Piedrahita, A. Giovanny, Guayacundo, D. Marcela, "Evaluation of Accelerometers as Inertial Navigation System for Mobile Robots," *IEEE 3rd Latin American Robotics Symp.*, pp. 84-90, 2006.

[36] Y. Wu, X. Hu, D. Hu, T. Li, J. Lian, "Strapdown Inertial Navigation System Algorithms Based on Dual Quaternions," *IEEE Trans. on Aerospace and Electronic Systems*, Vol 41, Issue 1, pp. 110-132, 2005.

[37] C. Dohyoung, L. G. Jang, G. P. Chan, W. P. Heung, "Strapdown INS Error Model for Multiposition Alignment," *IEEE Trans. on Aerospace and Electronic Systems*, Vol 32, Issue 4, pp. 1362-1366, 1996.

[38] J. Gaysse, "Using Fuzzy Logic in Low Cost Microcontroller to Increase Accelerometer Performance," *Proc. of the 2005 IEEE Mid-Summer Workshop on Soft Computing in Industrial Applications*, pp. 6-11, 2005.

[39] M. Park, Y. Gao, "Error Analysis and Stochastic Modeling of Low-cost MEMS Accelerometer," *Journal of Intelligent and Robotic Systems*, Vol. 46, Issue 1, pp. 27-41, 2006.

[40] J. Gao, P. Webb, N. Gindy, "Error Reduction for an Inertial-sensor-based Dynamic Parallel Kinematic Machine Positioning System," *Measurement Science & Technology*, Vol. 14, pp. 543-550, 2003.

[41] H. H. S. Liu, G. K. H. Pang, "Accelerometer for Mobile Robot Positioning," *IEEE Trans. on Industry Applications*, Vol 37, Issue 3, pp. 812-819, 2001.

[42] S. C. Lee, Y. C. Huang, "Innovative Estimation Method with Measurement Likelihood for All-Accelerometer Type Inertial Navigation System," *IEEE Trans. on Aerospace and Electronic Systems*, Vol. 38, Issue 1, pp. 339-346, 2002.

[43] H. J. Lin, "GPS/INS Integration without Gyro," *Position Location and Navigation Symp.*, pp. 159-164, 2004.

[44] K. P. Ying, M. F. Golnaraghi, "A Vector-Based Gyro-Free Inertial Navigation System by Integration Existing Accelerometer Network in a Passenger Vehicle," *Position Location and Navigation Symp.*, pp. 234-242, 2004.

[45] C. W. Tan, S. Park, "Design of Accelerometer-Based Inertial Navigation Systems," *IEEE Trans. on Instrumentation and Measurement*, Vol. 54, Issue 6, pp. 2520-2530, 2005.

[46] C. Y. Hung, C. M. Fang, S. C. Lee, "A Compensator to Advance Gyro-Free INS Precision," *Int. Journal of Control, Automation, and Systems*, Vol. 4, No. 3, pp. 351-358, 2006.

[47] N. Yazdi, F. Ayazi, K. Najafi, "Micromachined inertial sensors," *Proc. of the IEEE*, Vol. 86, Issue 8, pp. 1640-1659, 1998.

[48] Analog Devices Inc., "MEMS and Sensors," http://www.analog.com/en/cat/0,2878,764,00.html

[49] A. R. Schuler, A. Grammatikos, K. A. Fegler, "Measuring rotational motion with linear accelerometers," *IEEE Trans. on Aerospace and Electronic Systems*, Vol. 3, No. 3, pp. 465-471, 1967.

[50] J. H. Chen, S. C. Lee, D. B. DeBra, "Gyroscope free strapdown inertial measurement unit by six linear accelerometers," *Journal of Guidance, Control, and Dynamics AIAA*, Vol. 17, No. 2, pp. 286-290, 1994.

[51] Analog Devices Inc., "Data sheet of ADXL103," http://www.analog.com.

[52] R. G. Brown, P. Y. C. Hwang, "Introduction to Random Signals and Applied Kalman Filtering," John Wiley & Sons, Inc., 3rd Edition, 1997.

[53] H. Nomura, I. Hayashi, N. Wakami, "A Learning Method of Fuzzy Inference Rules by Descent Method," *IEEE Int. Conf. on Fuzzy Systems*, pp. 203-210, 1992.

[54] A. Habbi, M. Zelmat, "An Improved Self-Tuning Mechanism of Fuzzy Control by Gradient Descent Method," *Int. Symp. on Computational Intelligence and Intelligent Informatics*, pp. 93-97, 2007.